

CONSTRAINED OPTIMIZATION FOR LIQUID CRYSTAL EQUILIBRIA: EXTENDED RESULTS

J. H. ADLER, D. B. EMERSON, S. P. MACLACHLAN, T. A. MANTEUFFEL

Abstract. This paper investigates energy-minimization finite-element approaches for the computation of nematic liquid crystal equilibrium configurations. We compare the performance of these methods when the necessary unit-length constraint is enforced by either continuous Lagrange multipliers or a penalty functional. Building on previous work in [1, 2], the penalty method is derived and the linearizations within the nonlinear iteration are shown to be well-posed under certain assumptions. In addition, the paper discusses the effects of tailored trust-region methods and nested iteration for both formulations. Such methods are aimed at increasing the efficiency and robustness of each algorithms' nonlinear iterations. Three representative, free-elastic, equilibrium problems are considered to examine each method's performance. The first two configurations have analytical solutions and, therefore, convergence to the true solution is considered. The third problem considers more complicated boundary conditions, relevant in ongoing research, simulating surface nano-patterning. A multigrid approach is introduced and tested for a flexoelectrically coupled model to establish scalability for highly complicated applications. The Lagrange multiplier method is found to outperform the penalty method in a number of measures, trust regions are shown to improve robustness, and nested iteration proves highly effective at reducing computational costs.

Key words. nematic liquid crystals, finite-element methods, Newton linearization, energy optimization, penalty methods, trust regions, multigrid methods.

AMS subject classifications. 76A15, 65N30, 49M15, 65N22, 90C30, 65K10, 65N55

1. Introduction. Liquid crystals are substances that possess mesophases with characteristics spanning those of isotropic liquids and solid crystals. That is, liquid crystals are fluid, yet exhibit long-range structured ordering. This paper considers nematic liquid crystals which consist of rod-like molecules whose average pointwise orientation is represented by a vector, $\mathbf{n}(x, y, z) = (n_1, n_2, n_3)^T$. This orientation vector is known as the director and is assumed to be headless for nematics. Therefore, \mathbf{n} and $-\mathbf{n}$ are indistinguishable at any point in the domain Ω , due to molecular symmetry. An important constraint on the director vector field is that \mathbf{n} remain of unit length pointwise throughout Ω . Thorough overviews of liquid crystal physics are found in [12, 21, 51].

The deformable ordering of liquid crystal structures, coupled with the materials' birefringent and dielectric properties, has led to many important applications and discoveries, most famously in display technologies. Additional modern applications include nanoparticle organization [56], photorefractive cells [14], and liquid crystal elastomers designed to produce effective actuator devices such as light driven motors [58] and artificial muscles [53]. Numerical simulations of liquid crystal equilibrium configurations are used to optimize the performance of devices, analyze experiments, validate theory, and suggest the presence of new physical phenomenon [1, 4]. Many current technologies and experiments, including bistable devices [15, 40], require simulations with anisotropic physical constants on two- and three-dimensional domains.

To this end, a theoretically supported energy-minimization finite-element approach using Newton linearization and a Lagrange multiplier for the pointwise constraint was developed in [1, 2]. The approach effectively enforces the unit-length constraint while converging to energy-minimizing configurations. However, alternative approaches to efficiently impose unit-length conformance exist, such as the renormalized Newton method presented in [46]. Penalty methods have also been applied to liquid crystal equilibrium problems [4, 27, 30] and are utilized extensively to simplify

the Leslie-Ericksen equations [22, 34] in nematohydrodynamics simulations [37–39]. In addition, penalty methods are used for unit-length constraints in certain ferromagnetic problems [31].

In this paper, we focus on the Frank-Oseen free-elastic energy model and aim to compare the performance of techniques enforcing the unit-length constraint via Lagrange multipliers or with penalty methods employing augmentations to the free-elastic energy functional. The energy model and approaches are discussed in Section 2. In Section 3, well-posedness for the intermediate Newton linearizations that arise in the penalty method formulation is established. In addition to the incomplete Newton stepping discussed in [2], several tailored trust-region methods are investigated in Section 4. These trust-region approaches include one- and two-dimensional subspace minimization techniques [10, 11, 43]. A modified penalty method, which normalizes the director after each step, is also introduced in this section. The resulting algorithms are tested on three benchmark free-elastic problems in Section 5. Two of the problems have analytical solutions and one simulates nano-patterned boundary conditions relevant in ongoing research [4, 5]. In each of the experiments conducted, the Lagrange multiplier method outperforms the penalty approaches in a number of measures. Moreover, trust-regions are shown to improve convergence robustness and nested iteration proves exceptionally effective at reducing overall computational costs. In Section 6, a Braess-Sarazin-type multigrid scheme [3, 9] is introduced for a flexoelectrically coupled problem with nano-patterned boundary conditions, which demonstrates scalability for highly complicated models with coupled physics. Finally, Section 7 gives some concluding remarks, and future work is discussed.

2. Energy Model and Minimization. While a number of free-elastic energy models exist [20, 25, 51], we consider the Frank-Oseen free-elastic energy herein [51, 55]. The equilibrium, free-elastic energy is represented by an integral functional depending on deformations of the director field, \mathbf{n} . Liquid crystal samples favor stable configurations attaining minimal free energy. As in [1, 2], let K_i , $i = 1, 2, 3$, be the Frank constants [25] with $K_i \geq 0$ by Ericksen’s inequalities [23], and let

$$\mathbf{Z} = \kappa \mathbf{n} \otimes \mathbf{n} + (\mathbf{I} - \mathbf{n} \otimes \mathbf{n}) = \mathbf{I} - (1 - \kappa) \mathbf{n} \otimes \mathbf{n},$$

where $\kappa = K_2/K_3$. In general, we consider the case that $K_2, K_3 \neq 0$. Denote the classical $L^2(\Omega)$ inner product and norm as $\langle \cdot, \cdot \rangle_0$ and $\| \cdot \|_0$, respectively, and the standard Euclidean inner product and norm as (\cdot, \cdot) and $|\cdot|$. Throughout this paper, we assume the presence of Dirichlet boundary conditions or mixed Dirichlet and periodic boundary on a rectangular domain and, therefore, utilize the null Lagrangian simplification discussed in [1, 2, 51]. Thus, the Frank-Oseen free-elastic energy for a domain, Ω , is written

$$\int_{\Omega} w_F dV = \frac{1}{2} K_1 \|\nabla \cdot \mathbf{n}\|_0^2 + \frac{1}{2} K_3 \langle \mathbf{Z} \nabla \times \mathbf{n}, \nabla \times \mathbf{n} \rangle_0. \quad (2.1)$$

Note that if $\kappa = 1$, \mathbf{Z} is reduced to the identity and the energy becomes a Div-Curl system. The Frank constants may be determined experimentally for different liquid crystal types, are often anisotropic (i.e. $K_1 \neq K_2 \neq K_3$), and may depend on temperature [21, 28].

The one-constant approximation such that $K_1 = K_2 = K_3$, discussed in [1, 2, 51], is a widely applied simplification [13, 37, 45, 51, 57] of (2.1). This simplification significantly reduces the complexity of the free-elastic energy functional for both theoretical

analysis and computational simulation. However, it ignores anisotropic physical characteristics, which play major roles in liquid crystal phenomena [5,33]. Therefore, both the penalty and Lagrange multiplier methods considered here do not rely on such an assumption.

2.1. Penalty and Lagrange Multiplier Energy Minimization. The admissible equilibrium state for a liquid crystal sample is one that minimizes the system free energy in (2.1), subject to the local constraint $(\mathbf{n}, \mathbf{n}) = 1$. Here, we discuss two energy-minimization approaches, imposing this constraint via a penalty method or with Lagrange multipliers. To compute the free-energy minimizing configurations, we define the functional

$$\mathcal{F}(\mathbf{n}) = K_1 \|\nabla \cdot \mathbf{n}\|_0^2 + K_3 \langle \mathbf{Z} \nabla \times \mathbf{n}, \nabla \times \mathbf{n} \rangle_0. \quad (2.2)$$

Throughout this paper, we will make use of the spaces

$$\begin{aligned} H(\operatorname{div}, \Omega) &= \{\mathbf{v} \in L^2(\Omega)^3 : \nabla \cdot \mathbf{v} \in L^2(\Omega)\}, \\ H(\operatorname{curl}, \Omega) &= \{\mathbf{v} \in L^2(\Omega)^3 : \nabla \times \mathbf{v} \in L^2(\Omega)^3\}, \end{aligned}$$

as well as

$$\mathcal{H}^{DC}(\Omega) = \{\mathbf{v} \in H(\operatorname{div}, \Omega) \cap H(\operatorname{curl}, \Omega) : B(\mathbf{v}) = \mathbf{g}\},$$

with norm $\|\mathbf{v}\|_{DC}^2 = \|\mathbf{v}\|_0^2 + \|\nabla \cdot \mathbf{v}\|_0^2 + \|\nabla \times \mathbf{v}\|_0^2$ and appropriate boundary conditions $B(\mathbf{v}) = \mathbf{g}$. Here, we assume that \mathbf{g} satisfies appropriate compatibility conditions for the operator B . For example, if B represents full Dirichlet boundary conditions and Ω has a Lipschitz continuous boundary, it is assumed that $\mathbf{g} \in H^{\frac{1}{2}}(\partial\Omega)^3$ [26]. Furthermore, let $\mathcal{H}_0^{DC}(\Omega) = \{\mathbf{v} \in H(\operatorname{div}, \Omega) \cap H(\operatorname{curl}, \Omega) : B(\mathbf{v}) = \mathbf{0}\}$. Note that if Ω is a Lipschitz domain and B imposes full Dirichlet boundary conditions on all components of \mathbf{v} , then $\mathcal{H}_0^{DC}(\Omega) = H_0^1(\Omega)^3$ [26, Lemma 2.5].

2.2. Lagrange Multiplier Formulation. For this approach, the pointwise unit-length constraint is imposed by a continuous Lagrange multiplier. Following [1,2], the Lagrangian is defined as

$$\mathcal{L}(\mathbf{n}, \lambda) = \mathcal{F}(\mathbf{n}) + \int_{\Omega} \lambda(\mathbf{x})((\mathbf{n}, \mathbf{n}) - 1) dV,$$

where $\lambda \in L^2(\Omega)$. First-order optimality conditions, given by

$$\begin{aligned} \mathcal{L}_{\mathbf{n}}[\mathbf{v}] &= \frac{\partial}{\partial \mathbf{n}} \mathcal{L}(\mathbf{n}, \lambda)[\mathbf{v}] = 0, & \forall \mathbf{v} \in \mathcal{H}_0^{DC}(\Omega), \\ \mathcal{L}_{\lambda}[\gamma] &= \frac{\partial}{\partial \lambda} \mathcal{L}(\mathbf{n}, \lambda)[\gamma] = 0, & \forall \gamma \in L^2(\Omega), \end{aligned}$$

are derived and linearized to yield the Newton update equations

$$\begin{bmatrix} \mathcal{L}_{\mathbf{nn}} & \mathcal{L}_{\mathbf{n}\lambda} \\ \mathcal{L}_{\lambda\mathbf{n}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \delta \mathbf{n} \\ \delta \lambda \end{bmatrix} = - \begin{bmatrix} \mathcal{L}_{\mathbf{n}} \\ \mathcal{L}_{\lambda} \end{bmatrix}, \quad (2.3)$$

where each of the system components are evaluated at the current approximations \mathbf{n}_k and λ_k , while $\delta \mathbf{n} = \mathbf{n}_{k+1} - \mathbf{n}_k$ and $\delta \lambda = \lambda_{k+1} - \lambda_k$ are the desired updates to

these approximations. The matrix-vector multiplication indicates the direction that the derivatives in the Hessian are taken. That is,

$$\begin{aligned}\mathcal{L}_{\mathbf{nn}}[\mathbf{v}] \cdot \delta \mathbf{n} &= \frac{\partial}{\partial \mathbf{n}} (\mathcal{L}_{\mathbf{n}}(\mathbf{n}_k, \lambda_k)[\mathbf{v}]) [\delta \mathbf{n}], & \mathcal{L}_{\mathbf{n}\lambda}[\mathbf{v}] \cdot \delta \lambda &= \frac{\partial}{\partial \lambda} (\mathcal{L}_{\mathbf{n}}(\mathbf{n}_k, \lambda_k)[\mathbf{v}]) [\delta \lambda], \\ \mathcal{L}_{\lambda \mathbf{n}}[\gamma] \cdot \delta \mathbf{n} &= \frac{\partial}{\partial \mathbf{n}} (\mathcal{L}_{\lambda}(\mathbf{n}_k, \lambda_k)[\gamma]) [\delta \mathbf{n}],\end{aligned}$$

where the partials denote Gâteaux derivatives in the respective variables. The above system represents a linearized variational system for which we seek solutions $\delta \mathbf{n}$ and $\delta \lambda$. The complete system is found in [2].

2.3. Penalty Method Formulation. In order to define the penalty approach, the free-energy functional in (2.2) is augmented with a weighted, positive term,

$$\mathcal{P}(\mathbf{n}) = K_1 \langle \nabla \cdot \mathbf{n}, \nabla \cdot \mathbf{n} \rangle_0 + K_3 \langle \mathbf{Z} \nabla \times \mathbf{n}, \nabla \times \mathbf{n} \rangle_0 + \zeta \langle \mathbf{n} \cdot \mathbf{n} - 1, \mathbf{n} \cdot \mathbf{n} - 1 \rangle_0, \quad (2.4)$$

where $\zeta > 0$ represents a constant weight, penalizing deviations of the solution from the unit-length constraint. Thus, in the limit of large ζ values, unconstrained minimization of (2.4) is equivalent to the constrained minimization of (2.2). In order to minimize $\mathcal{P}(\mathbf{n})$, we compute the Gâteaux derivative of $\mathcal{P}(\mathbf{n})$ with respect to \mathbf{n} in the direction $\mathbf{v} \in \mathcal{H}_0^{DC}(\Omega)$. Hence, the first-order optimality condition is

$$\mathcal{P}_{\mathbf{n}}[\mathbf{v}] = \frac{\partial}{\partial \mathbf{n}} \mathcal{P}(\mathbf{n})[\mathbf{v}] = 0, \quad \forall \mathbf{v} \in \mathcal{H}_0^{DC}(\Omega).$$

Computation of this derivative yields the variational problem

$$\begin{aligned}\mathcal{P}_{\mathbf{n}}[\mathbf{v}] &= 2K_1 \langle \nabla \cdot \mathbf{n}, \nabla \cdot \mathbf{v} \rangle_0 + 2K_3 \langle \mathbf{Z} \nabla \times \mathbf{n}, \nabla \times \mathbf{v} \rangle_0 \\ &\quad + 2(K_2 - K_3) \langle \mathbf{n} \cdot \nabla \times \mathbf{n}, \mathbf{v} \cdot \nabla \times \mathbf{n} \rangle_0 + 4\zeta \langle \mathbf{v} \cdot \mathbf{n}, \mathbf{n} \cdot \mathbf{n} - 1 \rangle_0 = 0,\end{aligned}$$

for all $\mathbf{v} \in \mathcal{H}_0^{DC}(\Omega)$.

As with the Lagrangian formulation, the variational problem above contains nonlinearities. Therefore, Newton iterations are again applied, requiring computation of the second-order Gâteaux derivative with respect to \mathbf{n} . Let \mathbf{n}_k be the current approximation for \mathbf{n} and $\delta \mathbf{n} = \mathbf{n}_{k+1} - \mathbf{n}_k$ be the update that we seek to compute. Then, the Newton linearizations are written

$$\frac{\partial}{\partial \mathbf{n}} (\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]) [\delta \mathbf{n}] = -\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}], \quad \forall \mathbf{v} \in \mathcal{H}_0^{DC}(\Omega), \quad (2.5)$$

where

$$\begin{aligned}\frac{\partial}{\partial \mathbf{n}} (\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]) [\delta \mathbf{n}] &= 2K_1 \langle \nabla \cdot \delta \mathbf{n}, \nabla \cdot \mathbf{v} \rangle_0 + 2K_3 \langle \mathbf{Z}(\mathbf{n}_k) \nabla \times \delta \mathbf{n}, \nabla \times \mathbf{v} \rangle_0 \\ &\quad + 2(K_2 - K_3) \left(\langle \delta \mathbf{n} \cdot \nabla \times \mathbf{v}, \mathbf{n}_k \cdot \nabla \times \mathbf{n}_k \rangle_0 \right. \\ &\quad + \langle \mathbf{n}_k \cdot \nabla \times \mathbf{v}, \delta \mathbf{n} \cdot \nabla \times \mathbf{n}_k \rangle_0 + \langle \mathbf{n}_k \cdot \nabla \times \mathbf{n}_k, \mathbf{v} \cdot \nabla \times \delta \mathbf{n} \rangle_0 \\ &\quad + \langle \mathbf{n}_k \cdot \nabla \times \delta \mathbf{n}, \mathbf{v} \cdot \nabla \times \mathbf{n}_k \rangle_0 + \langle \delta \mathbf{n} \cdot \nabla \times \mathbf{n}_k, \mathbf{v} \cdot \nabla \times \mathbf{n}_k \rangle_0 \Big) \\ &\quad + 4\zeta (\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{v} \cdot \delta \mathbf{n} \rangle_0 + 2\langle \delta \mathbf{n} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0). \quad (2.6)\end{aligned}$$

Completing (2.5) with the above second-order derivative computation yields a linearized variational system. For each iteration, we compute $\delta \mathbf{n}$ satisfying (2.5) for all $\mathbf{v} \in \mathcal{H}_0^{DC}(\Omega)$ with the current approximation \mathbf{n}_k .

3. Well-Posedness of the Penalty Newton Linearizations. Existence and uniqueness of solutions to the discrete form of the linearization systems described in (2.3) is established in [2] under reasonable assumptions. Here, we adapt these results to the discrete form of the penalty method linearizations in Equation (2.5).

Let $a(\delta \mathbf{n}, \mathbf{v})$ denote the bilinear form defined in (2.6) for fixed \mathbf{n}_k and $F(\mathbf{v})$ be the linear functional on the right-hand side of the linearization in (2.5). Using finite elements to approximate the desired update, $\delta \mathbf{n}$, and considering a discrete space $V_h \subset \mathcal{H}_0^{DC}(\Omega)$ yields the discrete linearized system,

$$a(\delta \mathbf{n}_h, \mathbf{v}_h) = F(\mathbf{v}_h), \quad \forall \mathbf{v}_h \in V_h. \quad (3.1)$$

Throughout the rest of this section, the developed theory applies exclusively to discrete spaces. Therefore, except when necessary for clarity, we drop the subscript h along with the notation, $\delta \mathbf{n}$. For instance, we write $a(\mathbf{u}, \mathbf{v})$ to indicate the bilinear form in (3.1) operating on the discrete space $V_h \times V_h$. Furthermore, we refer to the following set of assumptions.

ASSUMPTION 3.1. *Consider an open bounded domain, Ω , with a Lipschitz-continuous boundary. Further, assume that there exist constants $0 < \alpha \leq 1 \leq \beta$, such that $\alpha \leq |\mathbf{n}_k|^2 \leq \beta$ and $\mathbf{Z}(\mathbf{n}_k(\mathbf{x}))$ remains uniformly symmetric positive definite (USPD) with lower and upper bounds on its Rayleigh quotient, η and Λ , respectively, as in [2, Lemma 2.1]. Finally, assume that Dirichlet boundary conditions are applied. While the assumption above, and the theory below, explicitly concern full Dirichlet boundary conditions, the theory is equally applicable to mixed Dirichlet and periodic boundary conditions on a rectangular domain.*

In order to establish well-posedness of (3.1), we show that the functional, $F(\mathbf{v})$, is continuous and that the bilinear form, $a(\mathbf{u}, \mathbf{v})$, is continuous and coercive. Decomposing the bilinear form, $a(\mathbf{u}, \mathbf{v})$, and the linear form, $F(\mathbf{v})$, into terms that contain the penalty term and those that do not,

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) &= \hat{a}(\mathbf{u}, \mathbf{v}) + 2\zeta(\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{v} \cdot \mathbf{u} \rangle_0 + 2\langle \mathbf{u} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0) \\ F(\mathbf{v}) &= \hat{F}(\mathbf{v}) + 2\zeta\langle \mathbf{v} \cdot \mathbf{n}_k, \mathbf{n}_k \cdot \mathbf{n}_k - 1 \rangle_0, \end{aligned}$$

we then extend the results in [2].

LEMMA 3.2. *Under Assumption 3.1, F is a bounded linear functional on V_h .*

Proof. From the bounds derived in [2, Lemma 3.6] and an application of the Cauchy-Schwarz inequality,

$$\begin{aligned} |F(\mathbf{v})| &\leq |\hat{F}(\mathbf{v})| + 2\zeta\|\mathbf{n}_k \cdot \mathbf{n}_k - 1\|_0\|\mathbf{n}_k \cdot \mathbf{v}\|_0 \\ &\leq C_F\|\mathbf{v}\|_{DC} + 2\zeta\|\mathbf{n}_k \cdot \mathbf{n}_k - 1\|_0\|\mathbf{n}_k \cdot \mathbf{v}\|_0, \end{aligned}$$

where C_F is a constant independent of mesh size, defined in [2]. Note that by assumption $\alpha \leq \mathbf{n}_k \cdot \mathbf{n}_k \leq \beta$, where $0 < \alpha \leq 1 \leq \beta$. Then, letting $C_\mu = \max(1 - \alpha, \beta - 1)$,

$$\|\mathbf{n}_k \cdot \mathbf{n}_k - 1\|_0^2 = \int_{\Omega} (\mathbf{n}_k \cdot \mathbf{n}_k - 1)^2 dV \leq C_\mu^2 \int_{\Omega} dV = C_\mu^2 |\Omega|.$$

Hence, $\|\mathbf{n}_k \cdot \mathbf{n}_k - 1\|_0 \leq C_\mu |\Omega|^{\frac{1}{2}}$. In addition,

$$\|\mathbf{n}_k \cdot \mathbf{v}\|_0 \leq \sqrt{\beta}\|\mathbf{v}\|_0 \leq \sqrt{\beta}\|\mathbf{v}\|_{DC}.$$

Thus,

$$|F(\mathbf{v})| \leq C_F\|\mathbf{v}\|_{DC} + 2\zeta C_\mu |\Omega|^{\frac{1}{2}} \sqrt{\beta}\|\mathbf{v}\|_{DC}.$$

□

LEMMA 3.3. *Under Assumption 3.1, $a(\mathbf{u}, \mathbf{v})$ is continuous.*

Proof. Using the bounds derived in [2, Lemma 3.7],

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) &\leq \hat{a}(\mathbf{u}, \mathbf{v}) + 2\zeta(|\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{u} \cdot \mathbf{v} \rangle_0| + 2|\langle \mathbf{u} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0|) \\ &\leq C_A \|\mathbf{u}\|_{DC} \|\mathbf{v}\|_{DC} + 2\zeta(|\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{u} \cdot \mathbf{v} \rangle_0| + 2|\langle \mathbf{u} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0|), \end{aligned}$$

where C_A is the continuity constant defined in [2]. Note that,

$$|\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{u} \cdot \mathbf{v} \rangle_0| = |\langle (\mathbf{n}_k \cdot \mathbf{n}_k - 1)\mathbf{u}, \mathbf{v} \rangle_0| \leq \|(\mathbf{n}_k \cdot \mathbf{n}_k - 1)\mathbf{u}\|_0 \|\mathbf{v}\|_0.$$

Furthermore,

$$\|(\mathbf{n}_k \cdot \mathbf{n}_k - 1)\mathbf{u}\|_0^2 = \int_{\Omega} (\mathbf{n}_k \cdot \mathbf{n}_k - 1)^2 (\mathbf{u} \cdot \mathbf{u}) dV \leq C_{\mu}^2 \|\mathbf{u}\|_0^2.$$

This implies that

$$\|(\mathbf{n}_k \cdot \mathbf{n}_k - 1)\mathbf{u}\|_0 \leq C_{\mu} \|\mathbf{u}\|_0,$$

and

$$|\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{u} \cdot \mathbf{v} \rangle_0| \leq C_{\mu} \|\mathbf{u}\|_{DC} \|\mathbf{v}\|_{DC}.$$

Noting that

$$|\langle \mathbf{u} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0| \leq \beta \|\mathbf{u}\|_{DC} \|\mathbf{v}\|_{DC},$$

we bound

$$a(\mathbf{u}, \mathbf{v}) \leq \left(C_A + 2\zeta(C_{\mu} + 2\beta) \right) \|\mathbf{u}\|_{DC} \|\mathbf{v}\|_{DC}.$$

□

Following the theory established in [2], two coercivity lemmas for $a(\mathbf{u}, \mathbf{v})$ are proved. The first proof addresses the case when $\kappa = 1$. The second considers coercivity when κ lies in a neighborhood of unity, $\kappa \in (1 - \epsilon_2, 1 + \epsilon_1)$. Let $\alpha_0 > 0$ be the coercivity constant from [2, Lemma 3.7].

LEMMA 3.4. *Under Assumption 3.1, if $\kappa = 1$ and $2\zeta|\alpha - 1| < \alpha_0$, there exists a $\beta_0 > 0$ such that $\beta_0 \|\mathbf{v}\|_{DC}^2 \leq a(\mathbf{v}, \mathbf{v})$ for all $\mathbf{v} \in V_h$.*

Proof.

$$a(\mathbf{v}, \mathbf{v}) = \hat{a}(\mathbf{v}, \mathbf{v}) + 2\zeta\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{v} \cdot \mathbf{v} \rangle_0 + 4\zeta\langle \mathbf{v} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0$$

Using the coercivity of $\hat{a}(\mathbf{v}, \mathbf{v})$ from [2, Lemma 3.7] and the fact that $\langle \mathbf{v} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0 \geq 0$,

$$\alpha_0 \|\mathbf{v}\|_{DC}^2 \leq \hat{a}(\mathbf{v}, \mathbf{v}) + 4\zeta\langle \mathbf{v} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0.$$

Observe that

$$\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{v} \cdot \mathbf{v} \rangle_0 = \int_{\Omega} (\mathbf{n}_k \cdot \mathbf{n}_k - 1)(\mathbf{v} \cdot \mathbf{v}) dV.$$

If $\alpha \leq \mathbf{n}_k \cdot \mathbf{n}_k \leq \beta$ for all $\mathbf{x} \in \Omega$ with $0 < \alpha \leq 1 \leq \beta$, then $(\alpha - 1) \leq 0$, and

$$\langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{v} \cdot \mathbf{v} \rangle_0 \geq (\alpha - 1) \int_{\Omega} \mathbf{v} \cdot \mathbf{v} dV \geq (\alpha - 1) \|\mathbf{v}\|_{DC}^2. \quad (3.2)$$

Letting $\beta_0 = \alpha_0 - 2\zeta|\alpha - 1|$,

$$\beta_0 \|\mathbf{v}\|_{DC}^2 \leq \hat{a}(\mathbf{v}, \mathbf{v}) + 4\zeta \langle \mathbf{v} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0 + 2\zeta \langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{v} \cdot \mathbf{v} \rangle_0.$$

Hence, if $2\zeta|\alpha - 1| < \alpha_0$, then $\beta_0 > 0$. \square

Therefore, $a(\mathbf{u}, \mathbf{v})$ is coercive for $\kappa = 1$ if ζ is not so large in comparison to the pointwise lower bound on the director length as to overwhelm α_0 .

As in [2], the assumption that $\kappa = 1$ can be loosened to include some anisotropy and retain coercivity of $a(\mathbf{u}, \mathbf{v})$. Let $C > 0$ such that $\|\mathbf{v}\|_0^2 \leq C(\|\nabla \cdot \mathbf{v}\|_0^2 + \|\nabla \times \mathbf{v}\|_0^2)$ (see [2]). Further, let $\alpha_1 > 0$ be defined as in the proof of [2, Lemma 3.8], where $K' = \min(K_1, \eta K_3)$ and $\alpha_1 = \frac{K'}{C+1}$. The following extends the results of [2, Lemma 3.8] to the penalty method.

LEMMA 3.5 (Small Data). *Under Assumption 3.1, if*

$$\beta_1 = \frac{\min(K_1, K_3)}{C+1} - 2\zeta|\alpha - 1| > 0,$$

there exists $\epsilon_1, \epsilon_2 > 0$, dependent on $\beta = \max |\mathbf{n}_k|^2$, such that for $\kappa \in (1 - \epsilon_2, 1 + \epsilon_1)$, $a(\mathbf{u}, \mathbf{v})$ is coercive.

Proof. Let

$$\begin{aligned} \tilde{a}(\mathbf{v}, \mathbf{v}) &= K_1 \langle \nabla \cdot \mathbf{v}, \nabla \cdot \mathbf{v} \rangle_0 + K_3 \langle \mathbf{Z}(\mathbf{n}_k) \nabla \times \mathbf{v}, \nabla \times \mathbf{v} \rangle_0 + 4\zeta \langle \mathbf{v} \cdot \mathbf{n}_k, \mathbf{v} \cdot \mathbf{n}_k \rangle_0 \\ &\quad + 2\zeta \langle \mathbf{n}_k \cdot \mathbf{n}_k - 1, \mathbf{v} \cdot \mathbf{v} \rangle_0. \end{aligned}$$

From the proof of [2, Lemma 3.8], the fact that $\zeta > 0$, and (3.2),

$$(\alpha_1 - 2\zeta|\alpha - 1|) \|\mathbf{v}\|_{DC}^2 \leq \tilde{a}(\mathbf{v}, \mathbf{v}). \quad (3.3)$$

The USPD lower bound for $\mathbf{Z}(\mathbf{n}_k)$, η , may depend on κ [2, Lemma 2.1]. Thus, the proof is split into three cases.

Case 1. $\kappa = 1 + \epsilon_1$, for $\epsilon_1 > 0$.

If this case holds, then $\eta = 1$. Hence, α_1 , defined for (3.3), is independent of κ . Since $K_2 - K_3 = K_3(\kappa - 1)$, the discrete bilinear form of (2.6) becomes

$$\begin{aligned} a(\mathbf{v}, \mathbf{v}) &= \tilde{a}(\mathbf{v}, \mathbf{v}) + \epsilon_1 K_3 \left(2 \langle \mathbf{v} \cdot \nabla \times \mathbf{v}, \mathbf{n}_k \cdot \nabla \times \mathbf{n}_k \rangle_0 + 2 \langle \mathbf{n}_k \cdot \nabla \times \mathbf{v}, \mathbf{v} \cdot \nabla \times \mathbf{n}_k \rangle_0 \right. \\ &\quad \left. + \langle \mathbf{v} \cdot \nabla \times \mathbf{n}_k, \mathbf{v} \cdot \nabla \times \mathbf{n}_k \rangle_0 \right). \end{aligned} \quad (3.4)$$

Observe that from (3.3),

$$(\alpha_1 - 2\zeta|\alpha - 1|) \leq \tilde{a}(\mathbf{v}, \mathbf{v}) + \epsilon_1 K_3 \langle \mathbf{v} \cdot \nabla \times \mathbf{n}_k, \mathbf{v} \cdot \nabla \times \mathbf{n}_k \rangle_0. \quad (3.5)$$

Consider the magnitude of the terms in (3.4) not bounded from below in (3.5), denoted as $\mathcal{G}(\mathbf{v}, \mathbf{v})$. As in the proof of Lemma 3.8 in [2],

$$|\mathcal{G}(\mathbf{v}, \mathbf{v})| \leq \epsilon_1 \alpha_3 \|\mathbf{v}\|_{DC}^2, \quad (3.6)$$

where α_3 is a constant defined in [2]. Utilizing (3.5) and (3.6),

$$a(\mathbf{v}, \mathbf{v}) \geq \alpha_1 \|\mathbf{v}\|_{DC}^2 - 2\zeta|\alpha - 1| \|\mathbf{v}\|_{DC}^2 - \epsilon_1 \alpha_3 \|\mathbf{v}\|_{DC}^2 = (\beta_1 - \epsilon_1 \alpha_3) \|\mathbf{v}\|_{DC}^2.$$

It is, thus, sufficient to have $\epsilon_1 < \beta_1/\alpha_3$, guaranteeing that $(\beta_1 - \epsilon_1 \alpha_3) > 0$.

Case 2. $\kappa = 1 - \epsilon_2 > 0$, for $\epsilon_2 > 0$, and $K_1 < K_3$.

Since $\kappa < 1$, $\eta = 1 + (\kappa - 1)\beta = (1 - \epsilon_2\beta)$. For $K_1 < K_3$, there exists an ϵ_2 small enough, such that $K_1 < (1 - \epsilon_2\beta)K_3$. This implies that, for small enough ϵ_2 ,

$$\alpha_1 = \frac{\min(K_1, (1 - \epsilon_2\beta)K_3)}{(C + 1)} = \frac{K_1}{(C + 1)}.$$

Therefore, α_1 is again independent of κ . Since $K_2 - K_3 = K_3(\kappa - 1)$, the discrete bilinear form of (2.6) becomes

$$\begin{aligned} a(\mathbf{v}, \mathbf{v}) = & \tilde{a}(\mathbf{v}, \mathbf{v}) - \epsilon_2 K_3 \left(2\langle \mathbf{v} \cdot \nabla \times \mathbf{v}, \mathbf{n}_k \cdot \nabla \times \mathbf{n}_k \rangle_0 + 2\langle \mathbf{n}_k \cdot \nabla \times \mathbf{v}, \mathbf{v} \cdot \nabla \times \mathbf{n}_k \rangle_0 \right. \\ & \left. + \langle \mathbf{v} \cdot \nabla \times \mathbf{n}_k, \mathbf{v} \cdot \nabla \times \mathbf{n}_k \rangle_0 \right). \end{aligned} \quad (3.7)$$

The terms of (3.7), not already bounded from below in (3.3), are bounded, utilizing [2, Lemma 3.8], as

$$|\mathcal{G}(\mathbf{v}, \mathbf{v})| \leq \epsilon_2 \alpha_4 \|\mathbf{v}\|_{DC}^2, \quad (3.8)$$

where α_4 is a constant defined in [2]. Using (3.3) and (3.8) implies that

$$a(\mathbf{v}, \mathbf{v}) \geq \alpha_1 \|\mathbf{v}\|_{DC}^2 - 2\zeta|\alpha - 1| \|\mathbf{v}\|_{DC}^2 - \epsilon_2 \alpha_4 \|\mathbf{v}\|_{DC}^2 \geq (\beta_1 - \epsilon_2 \alpha_4) \|\mathbf{v}\|_{DC}^2.$$

Thus, possibly requiring ϵ_2 to be even smaller, choose $\epsilon_2 < \beta_1/\alpha_4$, so that $(\beta_1 - \epsilon_2 \alpha_4) > 0$.

In the case that $\kappa < 1$, the additional restriction that $\beta < \frac{1}{1-\kappa}$ for \mathbf{Z} to be USPD is necessary, which implies that $\epsilon_2 \beta < 1$ is required. Therefore, for any fixed β , ϵ_2 must also be taken small enough to satisfy this condition. Hence,

$$\epsilon_2 < \min \left(\frac{\beta_1}{\alpha_4}, \frac{K_3 - K_1}{\beta K_3}, \frac{1}{\beta} \right).$$

Case 3. $\kappa = 1 - \epsilon_2 > 0$, for $\epsilon_2 > 0$, and $K_3 \leq K_1$.

Here, again, $\eta = (1 - \epsilon_2\beta)$. For this case, it is clear that $(1 - \epsilon_2\beta)K_3 < K_1$. Thus,

$$\alpha_1 = \frac{(1 - \epsilon_2\beta)K_3}{(C + 1)}.$$

Using the same α_4 as in the previous case and similar arguments,

$$\begin{aligned} a(\mathbf{v}, \mathbf{v}) & \geq \alpha_1 \|\mathbf{v}\|_{DC}^2 - 2\zeta|\alpha - 1| \|\mathbf{v}\|_{DC}^2 - \epsilon_2 \alpha_4 \|\mathbf{v}\|_{DC}^2 \\ & = \left(\frac{K_3}{C + 1} - 2\zeta|\alpha - 1| - \frac{\epsilon_2 \beta K_3}{C + 1} - \epsilon_2 \alpha_4 \right) \|\mathbf{v}\|_{DC}^2 \\ & \geq \left(\beta_1 - \frac{\epsilon_2 \beta K_3}{C + 1} - \epsilon_2 \alpha_4 \right) \|\mathbf{v}\|_{DC}^2. \end{aligned}$$

Hence, in order for $\left(\beta_1 - \frac{\epsilon_2 \beta K_3}{C + 1} - \epsilon_2 \alpha_4 \right) > 0$ to hold, it is necessary that

$$\epsilon_2 < \frac{\beta_1(C + 1)}{K_3 \beta + \alpha_4(C + 1)}.$$

Finally, ϵ_2 must still be chosen sufficiently small with respect to β such that $\epsilon_2\beta < 1$, as in Case 2. Therefore,

$$\epsilon_2 < \min \left(\frac{\beta_1(C+1)}{K_3\beta + \alpha_4(C+1)}, \frac{1}{\beta} \right).$$

□

Although the bounds on ϵ_1 and ϵ_2 are complicated, the only constant therein that depends on ζ is α_2 . The remainder of the constants are independent of ζ .

The above lemmas allow for the formulation of the following summary theorem.

THEOREM 3.6. *Under Assumption 3.1, if the conditions of Lemma 3.4 or Lemma 3.5 are satisfied, the discrete variational problem in (3.1) is well-posed.*

Proof. Lemmas 3.2 and 3.3 imply that $F(\mathbf{v})$ and $a(\mathbf{u}, \mathbf{v})$ are continuous, respectively. Lemmas 3.4 or 3.5 imply that $a(\mathbf{u}, \mathbf{v})$ is coercive. Therefore, by the Lax-Milgram Theorem [8], (3.1) is a well-posed discrete variational problem. □

Therefore, the discretization of the linearizations arising in the penalty method are always well-posed under the assumption of small anisotropy in the system coefficients and sufficient conformance to the unit-length constraint. However, the penalty parameter must be chosen appropriately to achieve accurate representation of the unit-length constraint. If ζ is too small, constraint conformance becomes poor and the functional minimum does not accurately represent the constrained minimum. Alternatively, if ζ is too large, the solvability of the intermediate variational systems degrades in two possible ways. The neighborhood admitting coercivity around $\kappa = 1$ shrinks and the system becomes increasingly ill-conditioned due to a decreasing coercivity constant, or coercivity is lost entirely and the possibility of non-invertible matrices appears. On the other hand, the proof of such well-posedness does not require establishing an inf-sup condition that necessitates subtle choices of finite-element spaces, as used for the Lagrange multiplier approach in [1, 2].

4. Robust Newton Step Methods. The Newton method applied to the Lagrange multiplier formulation discussed in [1, 2] employs naïve incomplete Newton stepping. That is, for a computed Newton update direction, $\delta \mathbf{n}$, a constant damping factor, $0 < \omega \leq 1$, is applied such that the new iterate is given as $\mathbf{n}_{k+1} = \mathbf{n}_k + \omega \delta \mathbf{n}$. Such an approach aims to improve convergence robustness when dealing with an inaccurate initial guess on coarse grids. However, this procedure may miss opportunities to take larger steps in “good” descent directions that effectively reduce the free energy.

Trust-region techniques are specifically designed to improve the robustness and efficiency of iterative procedures such as Newton’s method. Updates are confined to a neighborhood, known as a trust region, where the accuracy of the linearized first-order optimality conditions is “trusted”. These neighborhoods are expanded or contracted based on a measure of the model fidelity for a computed update. Significant research has produced both theoretical support and practical applications of such techniques [43]. This section discusses the use of constrained and unconstrained trust-region methods for the Lagrangian and penalty approaches discussed in Section 2. For a general overview of trust-region methods, see [43].

4.1. Trust-Region Approaches for the Penalty Formulation. Using the penalty functional in (2.4), the desired energy minimization is unconstrained. For this subsection, we denote the discretized forms of $\frac{\partial}{\partial \mathbf{n}} (\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]) [\delta \mathbf{n}]$ and $\mathcal{P}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]$ as U_k and \mathbf{f}_k , respectively. The quadratic model of the penalty functional, for a given

\mathbf{n}_k , is written

$$M_k(\delta\mathbf{n}) = \mathcal{P}(\mathbf{n}_k) + \mathbf{f}_k^T \delta\mathbf{n} + \frac{1}{2} \delta\mathbf{n}^T U_k \delta\mathbf{n}. \quad (4.1)$$

As a consequence of the well-posedness theory developed in Section 3, the matrix U_k is positive definite for each iteration. Therefore, we follow the methodology in [11, 48], computing steps by solving a trust-region minimization problem.

We seek an efficiently computable correction, $\delta\mathbf{n}$, that approximately minimizes the model in (4.1). In the following, we introduce two approaches to computing step length and direction for this problem. The performance of these techniques is vetted in the numerical experiments below.

Incomplete Newton stepping is equivalent to taking a small step in the descent direction, $-U_k^{-1}\mathbf{f}_k$. This is an effective means of finding energy minimizing solutions for both the penalty and Lagrangian methods. Therefore, in the first approach, a simple step selection technique is used in which the step is chosen satisfying the constrained minimization problem

$$\begin{aligned} \delta\mathbf{n}(\Delta_k) = \operatorname{argmin}\{ & \mathcal{P}(\mathbf{n}_k) + \mathbf{f}_k^T \delta\mathbf{n} + \frac{1}{2} \delta\mathbf{n}^T U_k \delta\mathbf{n} : \\ & |\delta\mathbf{n}| \leq \Delta_k, \delta\mathbf{n} = \mu U_k^{-1} \mathbf{f}_k \}, \end{aligned} \quad (4.2)$$

where Δ_k indicates the trust-region radius for iterate \mathbf{n}_k . Candidate solutions of (4.2) are easily computed to be $-U_k^{-1}\mathbf{f}_k$, the full Newton step, which may or may not be inside the trust region, and $\pm \frac{\Delta_k}{|U_k^{-1}\mathbf{f}_k|} U_k^{-1}\mathbf{f}_k$, representing steps to the trust region boundary.

An important aspect of trust-region methods is the adjustment of the trust-region radius and application of a computed step. This typically involves a measure of a computed step's merit. For a computed step, $\delta\mathbf{n}$, we compute the ratio,

$$\rho_k = \frac{\mathcal{P}(\mathbf{n}_k) - \mathcal{P}(\mathbf{n}_k + \delta\mathbf{n})}{M_k(\mathbf{0}) - M_k(\delta\mathbf{n})},$$

of the actual to the predicted reduction in \mathcal{P} due to the computed step. The closer ρ_k is to 1, the more accurately the quadratic model behavior matches that of the true functional.

If the ratio, ρ_k , is deemed acceptable, the step is applied and the trust region expands, remains static, or shrinks depending on the specific value of ρ_k . If ρ_k is too small, the step is rejected, the trust-region radius is shrunk, and the process repeated. To quantify, let $0 < \eta_3 < \eta_1 < \eta_2$ be positive constants, along with $0 < C_1 < 1 < C_3$. Further, let $\bar{\Delta}$ be a maximum limit on the trust-region size. Using these parameters, the specific decision trees for accepting the step, and subsequently adjusting the trust region, are given in Procedures 1 and 2, respectively.

Procedure 1: Solution update.	Procedure 2: TR size adjustment.
<pre> if $\rho_k > \eta_3$ then Accept step: $\mathbf{n}_{k+1} = \mathbf{n}_k + \delta \mathbf{n}$. else Reject step: $\mathbf{n}_{k+1} = \mathbf{n}_k$. end </pre>	<pre> if $\rho_k < \eta_1$ then Shrink region: $\Delta_{k+1} = C_1 \Delta_k$. else if $\rho_k > \eta_2$ and $\delta \mathbf{n} = \Delta_k$ then Expand region: $\Delta_{k+1} = \min(C_3 \Delta_k, \bar{\Delta})$. else Keep region constant: $\Delta_{k+1} = \Delta_k$. end </pre>

For our algorithm, if the components of the ratio, ρ_k , are very small and the computed step lies on the interior of the trust region, representing a full step towards satisfying the first-order optimality conditions, we choose to apply the step regardless of ρ_k and the trust region remains static. In this way, the trust-region minimization approach is used until we trust in the application of full Newton steps to obtain the first-order optimality conditions. A set of typical values for the trust-region parameters discussed above are listed in Table 5.1 and used in the numerical methods below.

A number of well-founded techniques improving trust-region step selection exist, including dogleg and two-dimensional (2D) subspace methods [11, 24, 43, 49]. Because the 2D-subspace method subsumes both the simple step selection approach above and dogleg methods, it is chosen as the alternative step selection computation here. Steps are computed by solving

$$\begin{aligned} \delta \mathbf{n}(\Delta_k) = \operatorname{argmin}\{ & \mathcal{P}(\mathbf{n}_k) + \mathbf{f}_k^T \delta \mathbf{n} + \frac{1}{2} \delta \mathbf{n}^T U_k \delta \mathbf{n} : \\ & |\delta \mathbf{n}| \leq \Delta_k, \delta \mathbf{n} = \mu_1 \mathbf{f}_k + \mu_2 U_k^{-1} \mathbf{f}_k \}. \end{aligned} \quad (4.3)$$

Again, the candidate solutions for (4.3) are efficiently computable, amounting to solving for the zeroes of a fourth-order polynomial.

4.1.1. A Renormalization Penalty Method. In addition to the standard penalty method discussed above, a modification is also considered in the numerical experiments below. Once the approximation to the solution has been updated with a computed and accepted step, the new approximation is renormalized at the finite-element nodes. That is, the updated approximation is projected onto the unit sphere at each finite-element node. This procedure is similar to that presented in [46] for a Lagrange multiplier formulation. There, the approach is derived within a nullspace method framework using the one constant approximation. Here, renormalization is applied to the penalty method, with and without trust regions and nested iteration, for anisotropic Frank constants.

This renormalization aims at improving unit-length conformance for solutions computed by the penalty method. The expectation is that this will lead to enhanced constraint conformance at lower penalty weights. However, unless the renormalization scaling is relatively uniform across nodes, the Newton direction may be significantly altered. Throughout this paper, this modification will be referred to as the “renormalization” penalty method.

4.2. A Trust-Region Method for the Lagrange Multiplier Formulation.

Applications of trust-region techniques to optimization problems with nonlinear constraints have also been developed. However, certain challenges arise in the theory and practical use of such methods [41]. Here, we consider existing trust-region approaches in the context of finite-element methods. For this subsection, let W_k be the matrix associated with a finite-element discretization of the second-order derivative of (2.2) (i.e., the functional without the Lagrange multiplier term), given by $\frac{\partial}{\partial \mathbf{n}} (\mathcal{F}_{\mathbf{n}}(\mathbf{n}_k)[\mathbf{v}]) [\delta \mathbf{n}]$. For the trust-region approach, write the constraint

$$c(\mathbf{n}) = \langle \mathbf{n} \cdot \mathbf{n} - 1, \mathbf{n} \cdot \mathbf{n} - 1 \rangle_0 = 0. \quad (4.4)$$

The Gâteaux derivative of (4.4) is

$$\frac{\partial}{\partial \mathbf{n}} c(\mathbf{n})[\mathbf{v}] = 4 \langle \mathbf{n} \cdot \mathbf{n} - 1, \mathbf{n} \cdot \mathbf{v} \rangle_0. \quad (4.5)$$

Finally, let \mathbf{c}_k be the column vector representing the finite-element discretized form of (4.5) at iterate \mathbf{n}_k .

One of the significant advantages of finite-element discretizations is the inherent sparsity of the resulting matrices. Trust-region algorithms in the Byrd-Omojokun family [10, 44, 52] require computation of the generally non-sparse matrix N_k , whose columns form an orthonormal basis for the orthogonal complement of \mathbf{c}_k , as well as the formation and inversion of the matrix $N_k^T W_k N_k$. In general, the matrix $N_k^T W_k N_k$ is not sparse and quite large, as W_k has dimension $m \times m$ and N_k is $m \times (m-1)$, where m is the number of discretization degrees of freedom for \mathbf{n} . Storage and computation with these dense matrices proves to be prohibitive, even on relatively small grids. Therefore, any advantages garnered by the use of these trust regions is outweighed by loss of the finite-element sparsity. Similarly, trust-region methods based on the fundamental work in [54] suffer from sparsity fill-in issues for large matrices in the context of finite-element methods.

To preserve sparsity properties, while still maintaining some advantages of a trust-region approach, we implement a simple trust-region method specifically fitted to the Lagrange multiplier formulation of the minimization problem. For the Lagrange multiplier approach in Section 2.2, we compute a Newton update direction as in [2], $\delta \chi = [\delta \mathbf{n} \ \delta \lambda]^T$. This update is meant to bring \mathbf{n}_k and λ_k closer to satisfying the first-order optimality conditions. Let $\mathcal{L}_0(\mathbf{n}_k, \lambda_k)$ represent the finite-element discretized form of the right-hand side of Equation (2.3) for \mathbf{n}_k and λ_k . Define the proportions w_k and w_{lim} , such that $0 < w_{\text{lim}} \leq w_k \leq 1$, where w_{lim} is a lower bound for w_k . For a given step, $w_k \delta \chi$, the expected change in $|\mathcal{L}_0(\mathbf{n}_k, \lambda_k)|$ is equal to $w_k |\mathcal{L}_0(\mathbf{n}_k, \lambda_k)|$. Therefore, we define the ratio

$$\rho_k = \frac{|\mathcal{L}_0(\mathbf{n}_k, \lambda_k)| - |\mathcal{L}_0(\mathbf{n}_k + w_k \delta \mathbf{n}, \lambda_k + w_k \delta \lambda)|}{w_k |\mathcal{L}_0(\mathbf{n}_k, \lambda_k)|}.$$

This ratio compares the change in the Lagrangian predicted by the linearized model to the actual change in the true Lagrangian for a computed step.

Let $0 < \eta_2 < \eta_1$ and $w_{\text{inc}}, w_{\text{dec}} \in (0, 1]$. Since w_k is a scaling factor, rather than a radius length, step selection and trust-region adjustment differ slightly from the procedures discussed above and are given in Procedures 3 and 4, respectively.

Procedure 3: Solution update.	Procedure 4: TR size adjustment.
<pre> if $\rho_k > \eta_2$ <i>or</i> $w_k = w_{lim}$ then Accept step: $[\mathbf{n}_{k+1} \ \lambda_{k+1}]^T =$ $[\mathbf{n}_k \ \lambda_k]^T + w_k \delta \chi$. else Reject step: $[\mathbf{n}_{k+1} \ \lambda_{k+1}]^T = [\mathbf{n}_k \ \lambda_k]^T$. end </pre>	<pre> if $\rho_k < \eta_2$ then Shrink region: $w_{k+1} = \max(w_{lim}, w_k - w_{dec})$. else if $\eta_2 < \rho_k < \eta_1$ then Keep region constant: $w_{k+1} = w_k$. else Expand region: $w_{k+1} = \min(w_k + w_{inc}, 1)$. end </pre>

5. Numerical Results. In this section, we compare the performance of the methods outlined above for three benchmark equilibrium problems. The general algorithm utilized by each method has three stages; see Algorithm 5. The outer stage implements nested iteration (NI) [50] where, at each level, the approximation to the solution is iteratively updated. These updates are computed via one of the methods described above. In general, the iteration stopping criterion, on a given level, is based on a set tolerance for the approximation's conformance to the first-order optimality conditions in the standard Euclidean l_2 -norm. For the renormalization penalty method, the Newton iteration tolerance is based on the reduction of the ratio of the energy from the previous step to the current step rather than conformance to the first-order optimality conditions. In the numerical experiments carried out below, both tolerances were held at 10^{-4} . The approximate solution is then transferred to a finer grid. In the current implementation, these finer grids represent uniform refinements of the initial coarse grid. However, adaptive refinement could also be performed.

The components of the variational problems in Equations (2.3) and (2.5) are discretized with finite elements on each grid. Both formulations use $Q_2 \times Q_2 \times Q_2$ elements for \mathbf{n} , while the Lagrange multiplier approach uses P_0 elements for λ , as in [1, 2]. In this section, the arising matrices are inverted using the UMFPACK LU decomposition [16–19]. In Section 6, we introduce an optimally scaling multigrid method with improved time to solution. The algorithm's discretizations and grid management are performed with the widely used deal.II finite-element and scientific computing library [6, 7].

Algorithm 5: General minimization algorithm with NI

```

0. Initialize solution approximation on coarse grid.
while Refinement limit not reached do
  | while Nonlinear iteration tolerance not satisfied do
  |   | 1. Assemble discrete components of System (2.3) or (2.5) on current
  |   |   grid,  $H$ .
  |   | 2. Compute correction to current approximation.
  |   | 3. Update current approximation.
  |   end
  | 4. Uniformly refine the grid to size  $h$ .
  | 5. Interpolate solution  $\mathbf{u}_H \rightarrow \mathbf{u}_h$ .
end

```

Each of the problems below is posed on a unit-square domain in the xy -plane, such that $\Omega = \{(x, y) \mid 0 \leq x, y \leq 1\}$. It is assumed that this domain represents a uniform slab. That is, the vector \mathbf{n} may have nonzero z -component but $\frac{\partial \mathbf{n}}{\partial z} = \mathbf{0}$. Dirichlet boundary conditions are applied at the y -edges and periodic boundary conditions are assumed at the boundaries $x = 0$ and $x = 1$. The experiments to follow consider an 8×8 coarse mesh ascending in 6 uniform refinements to a 512×512 mesh.

For the numerical experiments, each of the trust-region methods discussed above is applied. For the penalty trust-region methods, the initial trust region radius is set to Δ_{init} . At each refinement level, the trust-region radius is reset to Δ_{init} plus an incremental increase, Δ_{inc} , with a maximum of $\bar{\Delta}$. The Lagrangian trust-region approach sets the initial value of w_k to w_{init} . After each refinement, w_k is reset to w_{init} plus w_{lev} , up to a maximum of 1. These increments are due to the increasing accuracy of the iterates at each grid level. These constants are outlined in Tables 5.1 and 5.2

$\eta_1 = 0.25$	$\eta_2 = 0.75$	$\eta_3 = 0.125$	$C_1 = 0.5$
$C_3 = 1.3$	$\Delta_{\text{inc}} = 0.3$	$\bar{\Delta} = 100$	$\Delta_{\text{init}} = 0.3$

Table 5.1: Trust-region parameters for the penalty formulation.

$\eta_1 = 0.5$	$\eta_2 = 0.25$	$w_{\text{inc}} = 0.1$	$w_{\text{dec}} = 0.1$
$w_{\text{lev}} = 0.1$	$w_{\text{min}} = 0.1$	$w_{\text{init}} = 0.2$	—

Table 5.2: Trust-region parameters for the Lagrangian formulation.

The non-trust-region, incomplete Newton stepping approach is also performed for both formulations as a comparison benchmark with an initial $\omega = 0.2$, increasing by 0.2 at each refinement to a maximum of 1. The performance of each of these methods is then compared. In the results to follow, all reported free energies are computed using only the free elastic quantities without any augmentations, such as the penalty terms.

5.1. Twist Equilibrium Configuration. The first set of boundary conditions induce a classical twist equilibrium configuration [51]. For this experiment, and the tilt-twist experiment in the next subsection, let the general form of the solution be

$$\mathbf{n} = (\cos(\theta(y)) \cos(\phi(y)), \cos(\theta(y)) \sin(\phi(y)), \sin(\theta(y))). \quad (5.1)$$

Note that the known analytical solutions have a one-dimensional structure, but the numerical experiments below are full two-dimensional simulations. For the twist configuration, let $\theta_0 = \frac{\pi}{8}$. At the boundaries $\theta(0) = -\theta_0$, $\theta(1) = \theta_0$, and $\phi(0) = \phi(1) = 0$. The Frank constants for this problem are $K_1 = 1.0$, $K_2 = 1.2$, and $K_3 = 1.0$. The analytical equilibrium solution for these boundary conditions and Frank constants is derived, under a rotated coordinate system, in [51]. The solution is given by

$$\mathbf{n} = (\cos(\theta_0(2y - 1)), 0, \sin(\theta_0(2y - 1))),$$

with true free-elastic energy $2K_2\theta_0^2$. This corresponds to an expected free energy of 0.37011. The existence of an analytical solution for this problem allows for the computation of an L^2 -error for each computed approximation.

Type	Free Energy	L^2 -error	Min. Dev.	Max Dev.	Cost	TR Cost
Lagrangian	0.370110	2.076e-11	-1.43e-14	7.00e-15	1.350	1.340
Pen. $\zeta = 10^1$	0.358832	1.589e-02	-3.96e-02	-3.59e-05	1.371	1.354
Pen. $\zeta = 10^2$	0.368481	1.993e-03	-4.32e-03	-1.16e-05	1.376	1.355
Pen. $\zeta = 10^3$	0.369931	2.107e-04	-4.32e-04	-3.68e-06	1.440	1.418
Pen. $\zeta = 10^4$	0.370092	2.143e-05	-4.32e-05	-1.14e-06	1.448	1.420
Pen. $\zeta = 10^5$	0.370108	2.154e-06	-4.32e-06	-3.32e-07	1.447	1.426
Pen. $\zeta = 10^6$	0.370110	2.157e-07	-4.32e-07	-7.27e-08	—	1.436
Pen. $\zeta = 10^7$	0.370110	2.158e-08	-5.05e-08	-9.98e-09	—	1.465
Pen. $\zeta = 10^8$	0.370110	2.158e-09	-5.18e-09	-1.06e-09	—	1.516
Pen. $\zeta = 10^9$	0.370110	2.168e-10	-5.19e-10	-1.06e-10	—	1.639

Table 5.3: Statistics for the twist equilibrium solution with the different formulations and penalty weights. Included is the system free energy, the computed L^2 -error on the finest grid, and the minimum and maximum deviations from unit director length at the quadrature nodes. Approximations of the cost in WUs for the corresponding method with no trust regions and simple trust regions are included. Dashes in the columns indicate divergence.

Table 5.3 compares the performance of the Lagrange multiplier method to the penalty method without renormalization. The runs were performed with nested iteration and the approximate work, measured in terms of assembling and solving a single linearization step on a 512×512 grid, referred to as a work unit (WU), is given for the corresponding method with no trust regions and the simple trust region approaches, respectively. The work approximation is computed by summing the number of non-zeroes in each matrix across all grids and dividing by the number of non-zeroes in the (fixed) sparsity pattern at the finest level. The non-trust-region, incomplete Newton stepping discussed above diverged for penalty parameters of $\zeta = 10^6$ and greater. However, smaller damping parameters may yield convergence. Both penalty-method trust-region approaches converged without modification.

The table demonstrates the superior performance of the Lagrange multiplier method for this problem across all statistics with lower error, cost, and tighter conformance to the constraint. The penalty method does not match the free energy obtained by the Lagrangian formulation until reaching a penalty weight of 10^6 and, without trust regions, encounters divergence issues for these large penalty weights. While trust regions do not significantly reduce overall computations costs, Table 5.3 suggests that they significantly improve robustness.

The results in Tables 5.4 and 5.5 show the performance of the renormalization penalty method with and without trust regions. Table 5.5 provides additional statistics for the 2D-subspace minimization trust-region approach discussed in Table 5.4. For the twist equilibrium solution, the renormalization penalty method obtains better error values for smaller penalty weights than the unmodified penalty method. In Table 5.5, using the 2D-subspace minimization trust-region approach, we obtain an error of 3.592e-09 with a penalty weight of only $\zeta = 10^5$. Moreover, the minimum and maximum deviation of the director at the quadrature nodes is closer to that of the Lagrangian method. However, the performance improvements rely more heavily on the penalty parameter. While an error measure closer to the Lagrange multiplier formulation is achieved for $\zeta = 10^5$, performance degrades at $\zeta = 10^6$, with notable jumps in costs for all methods recorded in Table 5.4. The increases in error are due to

	No Trust Region		Simple Trust Region		2D Trust Region	
Type	L^2 -error	Cost	L^2 -error	Cost	L^2 -error	Cost
Pen. $\zeta = 10^1$	1.457e-02	1.338	1.457e-02	1.334	1.457e-02	1.334
Pen. $\zeta = 10^2$	8.932e-05	1.338	8.931e-05	1.334	8.931e-05	1.334
Pen. $\zeta = 10^3$	3.358e-06	1.339	3.357e-06	1.334	3.357e-06	1.335
Pen. $\zeta = 10^4$	1.523e-07	1.340	1.116e-07	1.336	1.116e-07	1.336
Pen. $\zeta = 10^5$	6.260e-08	8.113	3.595e-09	1.364	3.592e-09	1.340
Pen. $\zeta = 10^6$	6.356e-06	81.120	1.688e-02	73.052	1.098e-07	2.731

Table 5.4: A comparison of renormalization penalty methods, with and without trust-region approaches, for the twist solution. For each algorithm, the computed L^2 -error on the finest grid and an approximation of the cost in WUs is included.

the algorithm beginning to emphasize the unit-length constraint over proper director orientation. Correctly selecting the penalty weight represents a fundamental difficulty for this method.

Type	Free Energy	L^2 -error	Min. Dev.	Max Dev.	2D TR Cost
Pen. $\zeta = 10^1$	0.370168	1.457e-02	-4.58e-11	4.58e-11	1.334
Pen. $\zeta = 10^2$	0.370111	8.931e-05	-1.68e-11	1.68e-11	1.334
Pen. $\zeta = 10^3$	0.370110	3.357e-06	-5.18e-12	5.16e-12	1.335
Pen. $\zeta = 10^4$	0.370110	1.116e-07	-1.45e-12	1.43e-12	1.336
Pen. $\zeta = 10^5$	0.370110	3.592e-09	-3.16e-13	2.98e-13	1.340
Pen. $\zeta = 10^6$	0.370110	1.098e-07	-4.04e-14	2.20e-14	2.731

Table 5.5: Statistics for the twist equilibrium solution with different penalty weights. Here, the penalty method with renormalization and 2D-subspace minimization is considered. Included is the system free energy, the computed L^2 -error on the finest grid, the minimum and maximum deviations from unit director length at the quadrature nodes, and an approximation of the cost in WUs for the corresponding method.

Figure 5.1a displays the number of iterations required to reach the specified iteration tolerance within a nested iteration scheme alongside the final solution computed by the Lagrange multiplier formulation in Figure 5.1b. Counts for both the Lagrange multiplier approach and penalty formulation, with and without renormalization, for a penalty parameter $\zeta = 10^3$ are shown. In general, the trust-region methods significantly reduce iteration counts on the coarse grids. However, on the finer grids, this reduction is not sustained due to the efficiency of nested iteration. Because the improved iteration counts are confined to the coarsest grids, overall cost reduction is generally small. For example, the approximate cost for the Lagrange multiplier method was reduced very slightly from 1.350 WUs to 1.340 WUs, only resulting in a one second drop in overall time to solution.

Table 5.6 summarizes both the efficiency of nested iteration and highlights the strengths of certain applications of trust-region methods. For all of the constraint enforcement formulations, nested iteration offers very clear cost improvements. Coupling nested iteration with the Lagrange multiplier method for this problem is quite powerful, yielding the fastest overall run time and highest accuracy. Trust regions have a clear impact on time to solution in the absence of nested iteration but offer modest time to solution improvements when coupled with NI.

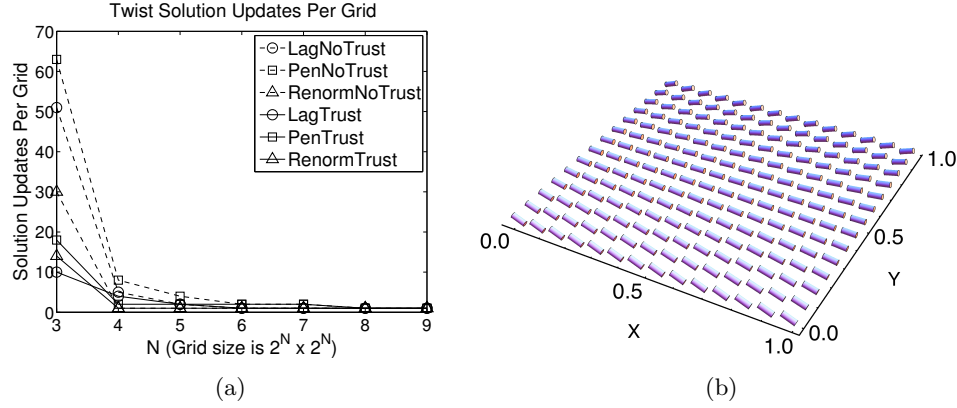


Fig. 5.1: (a) Number of iterations required to reach iteration tolerance for each method with NI. The penalty weight for the penalty formulation was $\zeta = 1000$. Only the 2D-subspace minimization trust-region approach is displayed, as the behavior of simple trust regions is similar. (b) The final computed solution for the Lagrangian formulation on a 512×512 mesh (restricted for visualization).

Lagrangian				Unmodified Penalty: $\zeta = 10^5$			
Method		Solve Cost	Run Time	Method		Solve Cost	Run Time
No NI	No TR	61	17,975s	No NI	No TR	142	41,013s
NI	No TR	1.350	550s	NI	No TR	1.474	593s
No NI	TR	10	3,071s	No NI	TR	63	18,425s
NI	TR	1.340	548s	No NI	TR 2D	64	19,287s
Renormalization Penalty: $\zeta = 10^5$				NI	TR	1.426	569s
Method		Solve Cost	Run Time	NI	TR 2D	1.424	574s
No NI	No TR	38	11,838s	Unmodified Penalty: $\zeta = 10^9$			
NI	No TR	8.113	2,272s	Method		Solve Cost	Run Time
No NI	TR	29	9,172s	No NI	No TR	—	—
No NI	TR 2D	32	10,147s	NI	No TR	—	—
NI	TR	1.364	585s	No NI	TR	1016	294,349s
NI	TR 2D	1.340	584s	No NI	TR 2D	1736	511,874s
				NI	TR	1.639	641s
				NI	TR 2D	1.958	764s

Table 5.6: Twist statistics comparison for NI and trust region combinations. The solve cost column displays an approximation of the work in WUs for the corresponding method. The overall time to solution is also presented. Dashes in the columns indicate divergence.

If the penalty method is used, pairing nested iteration with trust regions increases robustness and cost consistency. For example, the use of trust regions for the unmodified penalty method with $\zeta = 10^9$ overcomes prominent divergence issues. In addition to the improved error performance in Table 5.4, for $\zeta = 10^5$, the renormalization penalty method is generally faster than the unmodified penalty approach with the same penalty weight. The slightly slower overall run times when NI is paired with trust regions, in comparison with the unmodified penalty method, are due to the

work involved in normalizing the director after each iteration. As discussed above, a shortcoming of the renormalization penalty method is sensitivity to parameter choice.

5.2. Tilt-Twist Equilibrium Configuration. For this problem, \mathbf{n} retains the form in (5.1) and the same boundary conditions are applied with $\theta_0 = \frac{\pi}{4}$ and Frank constants of $K_1 = 1.0$, $K_2 = 3.0$, and $K_3 = 1.2$. Twist solutions incorporating a nonplanar tilt deviating from parallel alignment with the xz -plane are investigated in [35, 36]. It is shown that nonplanar twist solutions become available at a computable threshold. This threshold is satisfied for the chosen parameters. The analytical, energy-minimizing, tilt-twist solution is defined implicitly for a rotated coordinate system in [35, 51]. The associated analytical, free-elastic energy for the chosen parameters is 3.59294.

For the tilt-twist equilibrium solution, the incomplete Newton stepping approach converged for all of the penalty weights considered. Table 5.7 details the statistics for the unmodified penalty method compared with the Lagrange multiplier method. Again, the Lagrange multiplier method outperforms the penalty method in each category. The free energy of the Lagrange multiplier method is not obtained by the penalty method until ζ reaches 10^8 .

It should be noted that the behavior of the error for the Lagrangian method, as well as the penalty method for weights greater than 10^7 , is affected by the implicit definition of the true solution. The analytical solution for the tilt-twist equilibrium configuration is implicitly defined by a complicated set of equations, which are solved approximately at the appropriate quadrature points using Mathematica. Solving these equations involves successive root finding for complicated integral equations where the unknowns are limits of integration. Approximation error creates an artificial limit for the computed error at accuracies smaller than 10^{-7} .

Type	Free Energy	L^2 -error	Min. Dev.	Max Dev.	Cost	TR Cost
Lagrangian	3.59294	4.717e-07	-7.89e-10	7.88e-10	1.463	1.447
Pen. $\zeta = 10^1$	2.15620	4.403e-01	-4.78e-01	-2.62e-04	1.458	1.333
Pen. $\zeta = 10^2$	3.38037	4.597e-02	-5.01e-02	-1.17e-04	1.732	1.665
Pen. $\zeta = 10^3$	3.56953	4.565e-03	-4.97e-03	-3.88e-05	1.732	1.665
Pen. $\zeta = 10^4$	3.59052	4.606e-04	-5.00e-04	-1.21e-05	2.735	2.665
Pen. $\zeta = 10^5$	3.59269	4.590e-05	-5.00e-05	-3.56e-06	2.743	2.667
Pen. $\zeta = 10^6$	3.59291	4.253e-06	-5.01e-06	-8.05e-07	2.782	2.678
Pen. $\zeta = 10^7$	3.59293	2.735e-07	-5.83e-07	-1.14e-07	2.809	2.723
Pen. $\zeta = 10^8$	3.59294	4.340e-07	-6.00e-08	-1.22e-08	2.885	2.747
Pen. $\zeta = 10^9$	3.59294	4.676e-07	-6.01e-09	-1.24e-09	3.218	2.879

Table 5.7: Statistics for the tilt-twist equilibrium solution with the different formulations and penalty weights. Included is the system free energy, the computed L^2 -error on the finest grid, and the minimum and maximum deviations from unit director length at the quadrature nodes. Approximations of the cost in WUs for the corresponding method with no trust regions and simple trust regions are included.

Considering Tables 5.8 and 5.9, the renormalization penalty method does not perform as well as in the previous problem. Note that Table 5.9 details additional statistics for the 2D-subspace minimization trust-region approach discussed in Table 5.8. Compared to the unmodified penalty method, computational costs remain steadier, with the exception of the run without trust regions and a penalty weight of 10^6 ,

	No Trust Region		Simple Trust Region		2D Trust Region	
Type	L^2 -error	Cost	L^2 -error	Cost	L^2 -error	Cost
Pen. $\zeta = 10^1$	4.354e-01	1.384	4.319e-01	1.337	4.424e-01	1.377
Pen. $\zeta = 10^2$	3.691e-02	1.335	3.635e-02	1.333	3.578e-02	1.333
Pen. $\zeta = 10^3$	4.708e-03	1.335	4.533e-03	1.332	4.493e-03	1.332
Pen. $\zeta = 10^4$	1.085e-03	1.335	8.662e-04	1.332	8.536e-04	1.333
Pen. $\zeta = 10^5$	1.650e-03	1.336	9.487e-04	1.333	7.012e-04	1.333
Pen. $\zeta = 10^6$	9.414e-01	87.362	5.375e-04	1.341	7.344e-04	1.341

Table 5.8: A comparison of renormalization penalty methods, with and without trust-region approaches, for the tilt-twist solution. For each algorithm, the computed L^2 -error on the finest grid and an approximation of the cost in WUs is included.

and adherence to the unit-length constraint is improved. However, the method fails to reach an equivalent accuracy before performance degrades. As with the simpler twist problem, performance of the renormalization method is sensitive to an appropriate choice of penalty weight.

Type	Free Energy	L^2 -error	Min. Dev.	Max Dev.	2D TR Cost
Pen. $\zeta = 10^1$	3.92827	4.424e-01	-4.62e-09	4.62e-09	1.377
Pen. $\zeta = 10^2$	3.59611	3.578e-02	-1.15e-09	1.14e-09	1.333
Pen. $\zeta = 10^3$	3.59298	4.493e-03	-7.77e-10	7.76e-10	1.332
Pen. $\zeta = 10^4$	3.59294	8.536e-04	-7.87e-10	7.85e-10	1.333
Pen. $\zeta = 10^5$	3.59294	7.012e-04	-7.91e-10	7.90e-10	1.333
Pen. $\zeta = 10^6$	3.59294	7.344e-04	-7.87e-10	7.86e-10	1.341

Table 5.9: Statistics for the tilt-twist equilibrium solution with varying penalty weights. Here, the penalty method with renormalization and 2D-subspace minimization is shown. Included is the system free energy, the computed L^2 -error on the finest grid, the minimum and maximum deviations from unit director length at the quadrature nodes, and an approximation of the cost in WUs for the corresponding method.

The method with renormalization does find the true free energy at a lower penalty weight than the approach without renormalization. At a penalty weight of $\zeta = 10^4$, the penalty method without renormalization has a slightly lower error measure, but has not accurately matched the true energy. While the unmodified method more accurately resolves the orientation of the director in comparison with the renormalization method, it slightly shrinks the director length to attain the moderately smaller free energy.

Figure 5.2a presents similar behavior to Figure 5.1a, in that trust regions productively reduce the number of iterations on the coarsest grids but have less effect on iteration counts on the finest grids. This is again due to the efficiency of nested iteration. Figure 5.2b displays the solution computed by the Lagrange multiplier approach. For the unmodified penalty method with NI and a penalty weight of $\zeta = 10^9$, trust regions only reduce the computational cost from 3.218 WUs to 2.879 WUs. This results in only an 8.9% decrease in overall time to solution.

As shown in Table 5.10, improvements from the trust regions for the Lagrange multiplier method are minor, decreasing computational costs from 1.463 WUs to 1.447 WUs and reducing overall time to solution by only 0.82%. Here, the renormalization

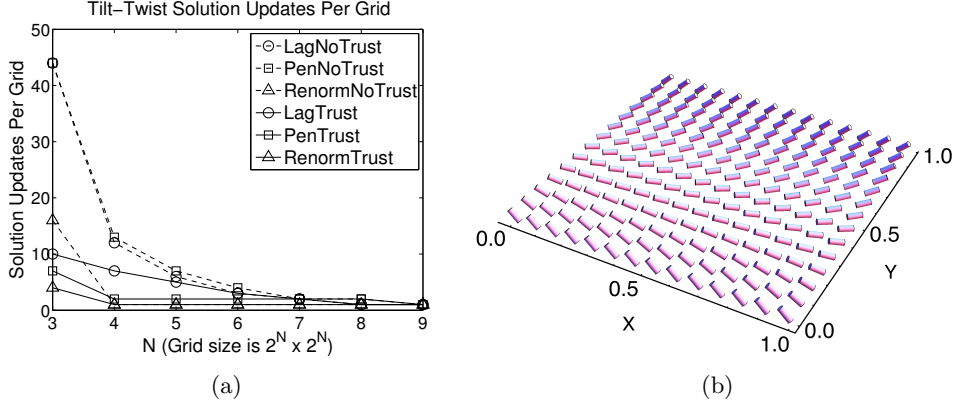


Fig. 5.2: (a) Number of iterations required to reach iteration tolerance for each method with NI. The penalty weight for the penalty formulation was $\zeta = 1000$. Only the 2D-subspace minimization trust-region approach is displayed, as the behavior of simple trust regions is similar. (b) The final computed solution for the Lagrangian formulation on a 512×512 mesh (restricted for visualization).

Lagrangian				Unmodified Penalty $\zeta = 10^5$			
Method		Solve Cost	Run Time	Method		Solve Cost	Run Time
No NI	No TR	33	9,853s	No NI	No TR	39	11,606s
NI	No TR	1.463	584s	NI	No TR	2.743	939s
No NI	TR	9	2,812s	No NI	TR	22	6,598s
NI	TR	1.447	579s	No NI	TR 2D	22	6,680s
Renormalization Penalty: $\zeta = 10^5$				NI	TR	2.667	920s
Method		Solve Cost	Run Time	NI	TR 2D	2.667	949s
No NI	No TR	16	5,119s				
NI	No TR	1.336	586s				
No NI	TR	18	5,658s				
No NI	TR 2D	18	5,817s				
NI	TR	1.333	575s				
NI	TR 2D	1.333	591s				

Table 5.10: Tilt-twist statistics comparison for NI and trust region combinations. The solve cost column displays an approximation of the work in WUs for the corresponding method. The overall time to solution is also presented.

penalty method is faster than the unmodified approach and, in some cases, even slightly outpaces the Lagrange multiplier formulation. However, as shown in Tables 5.7 and 5.9 the associated error convergence is not comparable. In Table 5.10, results for $\zeta = 10^9$ are not reported due to untenably large run times without nested iteration.

5.3. Nano Patterned Boundary Conditions. In this numerical experiment, we use Frank constants $K_1 = 1.0$, $K_2 = 0.62903$, and $K_3 = 1.32258$. The applied boundary conditions are the same as those used for the final experiment in [2]. Letting

$r = 0.25$ and $s = 0.95$, the boundary conditions are defined as

$$n_1 = 0, \quad (5.2)$$

$$n_2 = \cos(r(\pi + 2 \tan^{-1}(X_m) - 2 \tan^{-1}(X_p))), \quad (5.3)$$

$$n_3 = \sin(r(\pi + 2 \tan^{-1}(X_m) - 2 \tan^{-1}(X_p))), \quad (5.4)$$

where $X_m = \frac{-s \sin(2\pi(x+r))}{-s \cos(2\pi(x+r))-1}$ and $X_p = \frac{-s \sin(2\pi(x+r))}{-s \cos(2\pi(x+r))+1}$. These boundary conditions are pictured in Figure 5.3b. The result is a sharp transition from vertical nematics to planar-aligned rods followed by a rapid transition back to vertical alignment. Such boundary conditions produce configuration distortions throughout the interior of the domain and are important in physical applications [1, 4, 5]. Due to this complexity, no analytical solution currently exists.

The more complicated nature of the nano-patterned boundary conditions is reflected in the data of Table 5.11. The overall approximate costs for the methods with and without trust regions are larger than previous examples and the unit-length constraint is more difficult to capture. Nonetheless, the Lagrange multiplier method provides an accurate and cost effective approach. The penalty method without trust regions diverges for penalty weights greater than $\zeta = 10^4$. At higher penalty weights, even the trust-region approach suffers jumps in computational costs. At $\zeta = 10^9$, the system becomes over constrained and accuracy begins to degrade. Hence, results for this weight are not included.

Type	Free Energy	Min. Dev.	Max Dev.	Cost	TR Cost
Lagrangian	3.89001	-6.92e-05	5.89e-05	2.864	2.779
Pen. $\zeta = 10^1$	3.83657	-8.84e-02	1.96e-03	2.864	2.748
Pen. $\zeta = 10^2$	3.86896	-4.01e-02	4.40e-03	2.864	2.749
Pen. $\zeta = 10^3$	3.88331	-1.80e-02	7.32e-03	2.868	2.749
Pen. $\zeta = 10^4$	3.88819	-6.58e-03	5.81e-03	2.886	2.757
Pen. $\zeta = 10^5$	3.88965	-1.60e-03	2.01e-03	—	2.805
Pen. $\zeta = 10^6$	3.88996	-2.90e-04	4.55e-04	—	3.736
Pen. $\zeta = 10^7$	3.89001	-7.92e-05	1.01e-04	—	4.797
Pen. $\zeta = 10^8$	3.89001	-6.76e-05	5.83e-05	—	22.328

Table 5.11: Statistics for the nano-patterned equilibrium solution with the different formulations and penalty weights. Included is the system free energy and the minimum and maximum deviations from unit director length at the quadrature nodes. Approximations of the cost in WUs for the corresponding method with no trust regions and simple trust regions are included. Dashes in the columns indicate divergence.

As with the tilt-twist equilibrium solution, the renormalization penalty method approaches the Lagrangian formulation's free energy and unit-length constraint bounds earlier than the unmodified penalty method. It also yields a lower computational cost for most penalty weights. However, as was seen in the tilt-twist data, matching the energy earlier than the unmodified penalty approach does not directly indicate higher accuracy in resolving the correct orientation of the director. Moreover, in Table 5.12, divergence issues are apparent for the renormalization method at high penalty weights.

When considered with Table 5.11, Table 5.12 reinforces the conclusion that trust regions positively influence the robustness of penalty method approaches. While the simple trust-region approach works most effectively for the non-renormalization

penalty method, the 2D-subspace minimization approach is more favorable for the renormalization penalty formulation. Note that Table 5.13 details additional statistics for the 2D-subspace minimization trust-region approach discussed in Table 5.12.

Type	No Trust Region		Simple Trust Region		2D Trust Region	
	Free Energy	Cost	Free Energy	Cost	Free Energy	Cost
Pen. $\zeta = 10^1$	3.89319	1.680	3.89351	1.518	3.89308	1.686
Pen. $\zeta = 10^2$	3.89049	1.681	3.89051	1.666	3.89049	1.666
Pen. $\zeta = 10^3$	3.89006	1.682	3.89006	1.666	3.89006	1.666
Pen. $\zeta = 10^4$	3.89002	1.683	3.89002	1.669	3.89002	1.669
Pen. $\zeta = 10^5$	—	—	3.89001	2.133	3.89001	2.433
Pen. $\zeta = 10^6$	—	—	—	—	3.89001	5.418

Table 5.12: A comparison of renormalization penalty methods, with and without trust-region approaches, for the nano-pattern solution. For each algorithm, the computed free energy on the finest grid and an approximation of the cost in WUs is included. Dashes in the columns indicate divergence.

Type	Free Energy	Min. Dev.	Max Dev.	2D TR Cost
Pen. $\zeta = 10^1$	3.89308	-7.06e-05	6.02e-05	1.686
Pen. $\zeta = 10^2$	3.89049	-7.07e-05	6.02e-05	1.666
Pen. $\zeta = 10^3$	3.89006	-7.09e-05	6.01e-05	1.666
Pen. $\zeta = 10^4$	3.89002	-7.09e-05	6.01e-05	1.669
Pen. $\zeta = 10^5$	3.89001	-7.08e-05	6.00e-05	2.433
Pen. $\zeta = 10^6$	3.89001	-7.07e-05	5.98e-05	5.418

Table 5.13: Statistics for the nano equilibrium solution with the different formulations and penalty weights. Here, the penalty method with renormalization and 2D-subspace minimization is used. Included is the system free energy, the minimum and maximum deviations from unit director length at the quadrature nodes, and an approximation of the cost in WUs for the corresponding method.

Figure 5.3a displays the iteration counts as a function of grid size for both the Lagrange multiplier formulation and the penalty method, with and without renormalization, at a penalty weight of $\zeta = 10^3$. The solution computed by the Lagrange multiplier method is shown in Figure 5.3b. Similar to the previous problems, trust regions reduce iteration counts on the coarsest grids with reduced efficacy at the finer levels, due to NI. The cost savings from trust regions within a nested iteration scheme are slightly higher for this problem but persist as small improvements overall.

Table 5.14 reiterates the efficacy of nested iteration for efficient computation and trust regions for robustness. For this problem, Table 5.14 also shows that the renormalization penalty method with nested iteration and trust regions has a somewhat shorter overall run time than that of the Lagrange multiplier approach. Moreover, the renormalization approach matches the free energy and unit-length conformance of the Lagrangian formulation, see Tables 5.11 and 5.13. However, while the overall run time and approximate cost of the approach is slightly larger, the accuracy of the Lagrange multiplier formulation is expected to be much higher. For the Lagrange multiplier approach, the l_2 -norm of the first-order optimality conditions is 7.386e-13, whereas the same measure for the renormalization penalty method is 1.603e-02.

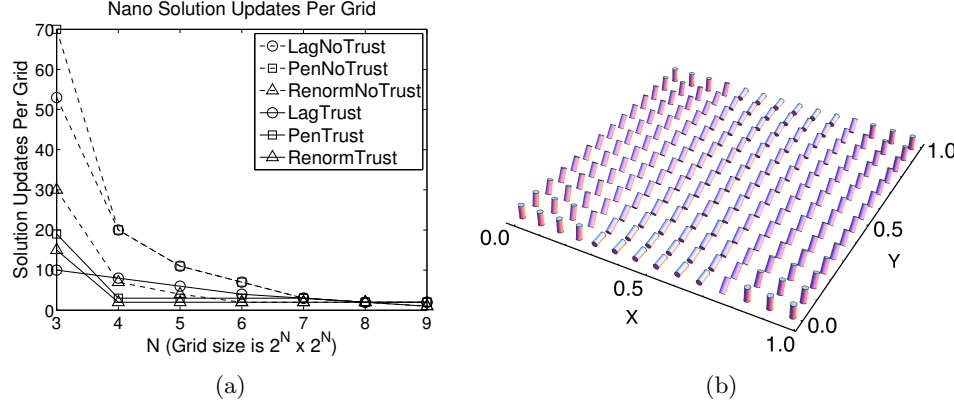


Fig. 5.3: (a) Number of iterations required to reach iteration tolerance for each method with NI. The penalty weight for the penalty formulation was $\zeta = 1000$. Only the 2D-subspace minimization trust-region approach is displayed, as the behavior of simple trust regions is similar. (b) The final computed solution for the Lagrangian formulation on a 512×512 mesh (restricted for visualization).

Lagrangian				Unmodified Penalty $\zeta = 10^9$			
Method		Solve Cost	Run Time	Method		Solve Cost	Run Time
No NI	No TR	63	18,861s	No NI	No TR	169	49,654s
NI	No TR	2.864	983s	NI	No TR	—	—
No NI	TR	10	3,113s	No NI	TR	73	21,415s
NI	TR	2.779	960s	No NI	TR 2D	75	22,366s
Renormalization Penalty: $\zeta = 10^5$				NI	TR	2.805	958s
Method		Solve Cost	Run Time	NI	TR 2D	3.530	1,202s
No NI	No TR	35	10,918s				
NI	No TR	—	—				
No NI	TR	32	9,893s				
No NI	TR 2D	34	10,976s				
NI	TR	2.133	789s				
NI	TR 2D	2.433	901s				

Table 5.14: Nano-pattern statistics comparison for NI and trust region combinations. The solve cost column displays an approximation of the work in WUs for the corresponding method. The overall time to solution is also presented. Dashes in the columns indicate divergence.

In all of the experiments above, the accuracy per unit cost of the Lagrange multiplier method convincingly outperforms that of either of the penalty methods. Moreover, the experimental results imply that nested iteration should be used when considering any of the methods. While trust regions offer very slight improvements in computation time, they readily improve robustness of the penalty method. Due to their limited cost, it would be advantageous to include them for either method. The simple trust-region approach works best for the unmodified penalty method with stopping tolerances based on the first-order optimality conditions, whereas the 2D-

subspace minimization trust regions are most effective for the renormalization penalty method with an energy reduction based stopping tolerance. Though larger penalty weights are generally necessary, the unmodified penalty method offers more consistent error reduction and performance with respect to an increasing weight.

6. Multigrid Solver. With the superiority of the Lagrange multiplier approach paired with nested iteration and trust regions established above, we demonstrate the full efficiency of the approach on a problem with electric and flexoelectric coupling as well as more complicated nano-patterned boundary conditions. In addition, a highly efficient, coupled multigrid method for the associated linear systems is introduced. The boundary conditions considered are a doubling of the nano-pattern described by Equations (5.2) - (5.4), such that the pattern contains a second strip parallel to the xy -plane. While there is no applied electric field, the curvature induced by the nano-patterning generates an internal electric field due to the flexoelectric properties of the liquid crystals [1, 42]. The flexoelectric free energy with appropriate boundary conditions is given as

$$\begin{aligned}\mathcal{F}(\mathbf{n}, \phi) = & K_1 \|\nabla \cdot \mathbf{n}\|_0^2 + K_3 \langle \mathbf{Z} \nabla \times \mathbf{n}, \nabla \times \mathbf{n} \rangle_0 \\ & - \epsilon_0 \epsilon_\perp \langle \nabla \phi, \nabla \phi \rangle_0 - \epsilon_0 \epsilon_a \langle \mathbf{n} \cdot \nabla \phi, \mathbf{n} \cdot \nabla \phi \rangle_0 \\ & + 2e_s \langle \nabla \cdot \mathbf{n}, \mathbf{n} \cdot \nabla \phi \rangle_0 + 2e_b \langle \mathbf{n} \times \nabla \times \mathbf{n}, \nabla \phi \rangle_0.\end{aligned}$$

The Frank constants used are $K_1 = K_3 = 1.0$ and $K_2 = 4.0$, and the electric constants are set to $\epsilon_\parallel = 7.0$, $\epsilon_\perp = 7.0$, $\epsilon_a = 0$, and $\epsilon_0 = 1.42809$. For the flexoelectric constants, using Rudquist notation [47], $e_s = 1.5$ and $e_b = -1.5$.

For a full derivation of the Lagrange multiplier approach extended to include electric and flexoelectric effects, see [1]. Using an electric potential, ϕ , the discretized and linearized system is written

$$\mathcal{M} \begin{bmatrix} \mathbf{n} \\ \phi \\ \lambda \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ B_1^T & -D & \mathbf{0} \\ B_2^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{n} \\ \phi \\ \lambda \end{bmatrix} = \begin{bmatrix} f_{\mathbf{n}} \\ f_\phi \\ f_\lambda \end{bmatrix}.$$

Define blocks of \mathcal{M} as

$$\hat{A} = \begin{bmatrix} A & B_1 \\ B_1^T & -D \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} B_2 \\ \mathbf{0} \end{bmatrix}. \quad (6.1)$$

Furthermore, let $\hat{u} = [\mathbf{n} \ \phi]^T$ and $\hat{f}_{\hat{u}} = [f_{\mathbf{n}} \ f_\phi]^T$. With these block definitions, Braess-Sarazin relaxation, originally formulated in [9] for Stokes flows, is used and takes the form

$$\begin{bmatrix} \hat{u}_{k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} \hat{u}_k \\ \lambda_k \end{bmatrix} + \begin{bmatrix} \gamma_b R & \hat{B} \\ \hat{B}^T & \mathbf{0} \end{bmatrix}^{-1} \left(\begin{bmatrix} \hat{f}_{\hat{u}} \\ f_\lambda \end{bmatrix} - \begin{bmatrix} \hat{A} & \hat{B} \\ \hat{B}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{u}_k \\ \lambda_k \end{bmatrix} \right), \quad (6.2)$$

where R is an appropriate preconditioner for \hat{A} and γ_b is a weighting parameter. For the multigrid approach, the matrix,

$$\begin{bmatrix} \gamma_b R & \hat{B} \\ \hat{B}^T & \mathbf{0} \end{bmatrix},$$

is only approximately inverted, as in [3]. We use Q_2 elements for both \mathbf{n} and ϕ , and, hence, the degrees of freedom for the components of \mathbf{n} and ϕ are collocated. As

suggested in [3], we construct the approximation, R , by extracting 4×4 -blocks of \hat{A} corresponding to the nodally collocated degrees of freedom for \mathbf{n} and ϕ . With careful permutation of the degrees of freedom in Equation (6.2), R becomes a block-diagonal matrix consisting of these 4×4 collocation blocks.

Comparative studies of Braess-Sarazin-type relaxation schemes in a multigrid framework for Stokes flows are found in [32], while numerical studies of their extension to multigrid methods for magnetohydrodynamic equations are performed in [3]. Here, as part of the underlying multigrid method, we use standard finite-element interpolation operators and Galerkin coarsening.

We first focus on determining the optimal value of the relaxation parameter. Here, and in the subsequent runs, the multigrid convergence tolerance, which is based on a ratio of the current solution's residual to that of the initial guess, remains fixed at 10^{-6} for each grid level and Newton step. The parameter γ_b was varied from 1.10 to 2.00 in increments of 0.05. Displayed in Figure 6.1a are the multigrid iteration counts, averaged over Newton iterations, for a 512×512 grid with respect to varying values of γ_b . The parameter study suggests that a γ_b value of 1.20 is optimal for convergence and, thus, we use this value. It is interesting to note that the iteration counts are relatively insensitive to increases in γ_b above 1.20.

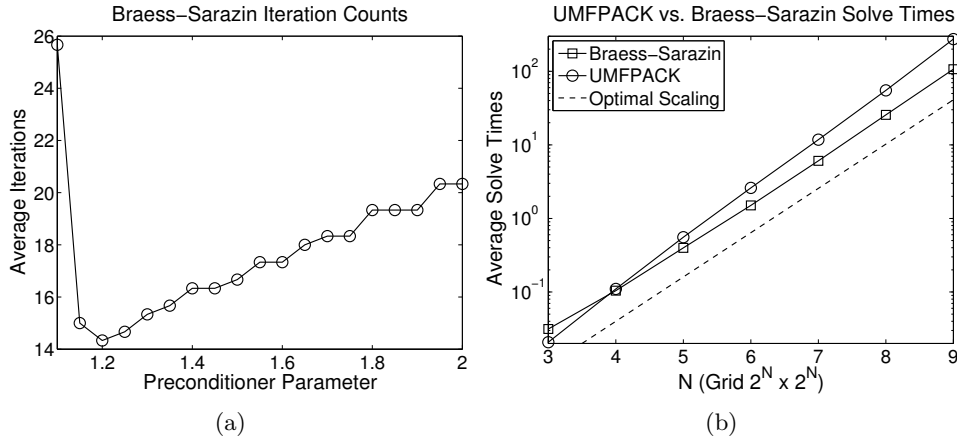


Fig. 6.1: (a) The average number of multigrid iterations for varying γ_b on a 512×512 grid. (b) The average time to solution for the Braess-Sarazin multigrid scheme compared to the UMFPACK direct solver.

Figure 6.1b exhibits average total setup and solve times for grid sizes from 8×8 to 512×512 for the UMFPACK direct solver and the Braess-Sarazin-type multigrid scheme. Using Q_2 - Q_2 - P_0 elements for \mathbf{n} , ϕ , and λ , respectively, the matrices on the 512×512 grid are of dimension $4,464,644 \times 4,464,644$ with 289,969,900 nonzero entries. Here, the Braess-Sarazin multigrid scheme is scaling optimally with the grid size. Furthermore, there is a clear timing crossover at the 16×16 mesh, at which point Braess-Sarazin becomes the faster solver. This timing intersection occurs considerably earlier than the Vanka-type relaxation scheme discussed in [1].

Table 6.1 details a comparison of the UMFPACK direct solver's performance to that of the Braess-Sarazin-type multigrid scheme. With and without trust regions, the computed free energy between the two solvers is identically 16.413. However,

Trust-Region Type	UMFPACK Solve		Braess-Sarazin Solve	
	None	Simple	None	Simple
System Assembly Time	136.1s	131.3s	136.8s	131.8s
Data Conversion Time	—	—	136.9s	132.6s
Linear Setup/Solve Time	1053.2s	1035.2s	436.8s	425.2s
Memory/Output Time	284.7s	283.2s	305.8s	302.6s
Total Time	1474.0s	1449.7s	1016.3s	992.2s

Table 6.1: A comparison of timing breakdowns for runs using the UMFPACK direct solver or the Braess-Sarazin multigrid scheme. Each solver is run with and without trust regions.

Braess-Sarazin-type relaxation reduces overall runtime by nearly 33%. This speed up is most notable when considering the fact that overall runtime for the multigrid solver experiments includes porting variables to types compatible with the Trilinos computational library [29] and computing collocation information for the Braess-Sarazin-type relaxation. Using the Lagrange multiplier formulation, nested iteration, multigrid, and trust regions, we obtain a robust and efficient algorithm.

7. Summary and Future Work. We have discussed three approaches for imposing the pointwise unit-length constraint required by the Frank-Oseen free-energy model for nematic liquid crystals, including a Lagrange multiplier formulation and a penalty method, with and without renormalization. Theory establishing the well-posedness of the intermediate discrete linearization systems for the penalty method was presented. Such theory parallels the theory establishing the well-posedness of the Lagrange multiplier linearization systems proved in [2]. In addition, trust-region methods specifically tailored to each of the formulations were discussed and implemented.

The ensuing algorithms were compared for three benchmark equilibrium problems. The experiments suggested that the Lagrange multiplier method coupled with nested iteration is the most accurate and efficient approach for enforcing the unit-length constraint. Trust-region schemes were shown to increase overall robustness with very little extra cost and, therefore, should be considered when using either constraint formulation. Furthermore, nested iteration was observed to be exceedingly effective at reducing computational costs for all problems and methods. Finally, a Braess-Sarazin-type multigrid method, based on work in [3], was introduced for the Lagrange multiplier formulation and successfully applied to a highly difficult liquid crystal problem with electric and flexoelectric coupling. The method quickly and accurately computed the expected equilibrium configuration.

The current implementation utilizes uniform grid refinement to build the nested iteration hierarchy of grids. Future work will include study of adaptive refinement techniques. Because the energy minimization formulation does not yield an obvious a posteriori error estimator, new techniques will be explored to flag cells for refinement. With an accurate and efficient approach for liquid crystal equilibrium configurations established, research into the application of energy-minimization finite-element approaches to liquid crystal flow problems will also be undertaken.

Acknowledgments. The authors would like to thank Professor Timothy Ather-ton for his useful suggestions and contributions. We would also like to thank Thomas Benson for allowing us to use and adapt his code, as well as his helpful contributions.

REFERENCES

- [1] J.H. Adler, T.J. Atherton, T.R. Benson, D.B. Emerson, and S.P. MacLachlan. Energy minimization for liquid crystal equilibrium with electric and flexoelectric effects. *Under Review*, 2014.
- [2] J.H. Adler, T.J. Atherton, D.B. Emerson, and S.P. MacLachlan. An energy-minimization finite-element approach for the Frank-Oseen model of nematic liquid crystals. *Under Review*, 2014.
- [3] J.H. Adler, T.R. Benson, E.C. Cyr, S.P. MacLachlan, and R.S. Tuminaro. Monolithic multigrid methods for 2D resistive magnetohydrodynamics. *In Preparation*, 2014.
- [4] T.J. Atherton and J.H. Adler. Competition of elasticity and flexoelectricity for bistable alignment of nematic liquid crystals on patterned surfaces. *Phys. Rev. E*, 86, 2012.
- [5] T.J. Atherton and J.R. Sambles. Orientational transition in a nematic liquid crystal at a patterned surface. *Phys. Rev. E*, 74, 2006.
- [6] W. Bangerth, R. Hartmann, and G. Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Trans. Math. Softw.*, 33(4):24/1–24/27, 2007.
- [7] W. Bangerth, T. Heister, G. Kanschat, et al. deal.II *Differential Equations Analysis Library, Technical Reference*. <http://www.dealii.org>.
- [8] D. Braess. *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics*. Cambridge University Press, 1997.
- [9] D. Braess and R. Sarazin. An efficient smoother for the stokes problem. *Appl. Numer. Math.*, 23:3–19, 1997.
- [10] R.H. Byrd, R.B. Schnabel, and G.A. Shultz. A trust region algorithm for nonlinearly constrained optimization. *SIAM J. Numer. Anal.*, 24(5):1152–1170, Oct. 1987.
- [11] R.H. Byrd, R.B. Schnabel, and G.A. Shultz. Approximate solution of the trust region problem by minimization over two-dimensional subspaces. *Math. Programming*, 40:247–263, 1988.
- [12] S. Chandrasekhar. *Liquid Crystals*. Cambridge University Press, Cambridge, 2nd edition, 1992.
- [13] R. Cohen, R. Hardt, D. Kinderlehrer, S. Lin, and M. Luskin. Minimum energy configurations for liquid crystals: Computational results. In *Theory and Applications of Liquid Crystals*, volume 5 of *The IMA Volumes in Mathematics and Its Applications*, pages 99–121. Springer-Verlag, 1987.
- [14] K.R. Daly, G. D’Alessandro, and M. Kaczmarek. Regime independent coupled-wave equations in anisotropic photorefractive media. *Appl. Phys. B: Lasers Opt.*, 95:589–596, 2009.
- [15] A.J. Davidson and N.J. Mottram. Flexoelectric switching in a bistable nematic device. *Phys. Rev. E*, 65(5), May 2002.
- [16] T.A. Davis. Algorithm 832: UMFPACK, an unsymmetric-pattern multifrontal method. *ACM Trans. Math. Softw.*, 30(2):196–199, 2004.
- [17] T.A. Davis. A column pre-ordering strategy for the unsymmetric-pattern multifrontal method. *ACM Trans. Math. Softw.*, 30(2):165–195, 2004.
- [18] T.A. Davis and I.S. Duff. An unsymmetric-pattern multifrontal method for sparse LU factorization. *SIAM J. Matrix Anal. Appl.*, 18(1):140–158, 1997.
- [19] T.A. Davis and I.S. Duff. A combine unifrontal/multifrontal method for unsymmetric sparse matrices. *ACM Trans. Math. Softw.*, 25(1):1–19, 1999.
- [20] T.A. Davis and E.C. Gartland Jr. Finite element analysis of the Landau-de Gennes minimization problem for liquid crystals. *SIAM J. Numer. Anal.*, 35(1):336–362, 1998.
- [21] P.G. de Gennes and J. Prost. *The Physics of Liquid Crystals*. Clarendon Press, Oxford, 2nd edition, 1993.
- [22] J. Ericksen. Continuum theory of nematic liquid crystals. *Res. Mechanica*, 21:381–392, 1987.
- [23] J.L. Ericksen. Inequalities in liquid crystal theory. *Phys. Fluids*, 9:1205–1207, 1966.
- [24] R. Fletcher. *Practical Methods of Optimization*, volume 1. John Wiley and Sons, Inc., New York, Brisbane, and Toronto, 1980.
- [25] F.C. Frank. On the theory of liquid crystals. *Discuss. Faraday Soc.*, 25:19–28, 1958.
- [26] V. Girault and P. Raviart. *Finite Element Methods for Navier-Stokes Equations*. Springer-Verlag, Germany, 1986.
- [27] R. Glowinski, P. Lin, and X. B. Pan. An operator-splitting method for a liquid crystal model. *Comput. Phys. Comm.*, 152(3):242–252, 2003.
- [28] I. Haller. Elastic constants of the nematic liquid crystalline phase of p-Methoxybenzylidene-p-n-Butylaniline (MBBA). *J. Chem. Phys.*, 57:1400–1405, 1972.
- [29] M.A. Heroux, R.A. Bartlett, V.E. Howle, R.J. Hoekstra, J.J. Hu, T.G. Kolda, R.B. Lehoucq, K.R. Long, R.P. Pawlowski, E.T. Phipps, A.G. Salinger, H.K. Thornquist, R.S. Tuminaro, J.M. Willenbring, A. Williams, and K.S. Stanley. An overview of the trilinos project. *ACM Trans. Math. Softw.*, 31(3):397–423, 2005.

- [30] Q. Hu and L. Yuan. A Newton-penalty method for a simplified liquid crystal model. *Adv. Comput. Math.*, 40(1):201–244, 2014.
- [31] M. Kružík and A. Prohl. Recent developments in the modeling, analysis, and numerics of ferromagnetism. *SIAM Rev.*, 48(3):439–483, 2006.
- [32] M. Larin and A. Reusken. A comparative study of efficient iterative solvers for generalized Stokes equations. *Numer. Linear Algebra Appl.*, 15:13–34, 2008.
- [33] B.W. Lee and N.A. Clark. Alignment of liquid crystals with patterned isotropic surfaces. *Science*, 291(5513):2576–2580, March 2001.
- [34] F. Leslie. Theory of flow phenomenon in liquid crystals. In *The Theory of Liquid Crystals*, volume 4, pages 1–81. Academic Press, 1979.
- [35] F.M. Leslie. Distorted twisted orientation patterns in nematic liquid crystals. *Pramana, Suppl. No.*, 1:41–55, 1975.
- [36] F.M. Leslie. Some topics in equilibrium theory of liquid crystals. In J.L. Ericksen and D. Kinderlehrer, editors, *Theory and Applications of Liquid Crystals*, pages 211–234. Springer-Verlag, New York, 1987.
- [37] P. Lin, C. Liu, and H. Zhang. An energy law preserving C0 finite element scheme for simulating the kinematic effects in liquid crystal flow dynamics. *J. Comp. Phys.*, (227):1411–1427, 2007.
- [38] C. Liu and N.J. Walkington. Mixed methods for the approximation of liquid crystal flows. *ESAIM Math. Model. Numer. Anal.*, 36(2):205–222, 2002.
- [39] C. Liu, H. Zhang, and S. Zhang. Numerical simulations of hydrodynamics of nematic liquid crystals: Effects of kinematic transports. *Commun. Comput. Phys.*, 9(4):974–993, 2011.
- [40] A. Majumdar, C.J.P. Newton, J.M. Robbins, and M. Zyskin. Topology and bistability in liquid crystal devices. *Phys. Rev. E*, 75, 2007.
- [41] N. Maratos. *Exact Penalty Function Algorithms for Finite Dimensional and Control Optimization*. PhD thesis, Univ. of London, London, 1978.
- [42] R.B. Meyer. Piezoelectric effects in liquid crystal. *Physical Review Letters*, 22(18), 1969.
- [43] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer, New York, 1999.
- [44] E.O. Omojokun. *Trust Region Algorithms for Optimization with Nonlinear Equality and Inequality Constraints*. PhD thesis, University of Colorado, Boulder, 1989.
- [45] A. Ramage and E.C. Gartland Jr. A preconditioned nullspace method for liquid crystal director modeling. *SIAM J. Sci. Comput.*, 35(1):B226–B247, 2013.
- [46] A. Ramage and E.C. Gartland Jr. A renormalized Newton method for liquid crystal director models with pointwise unit-vector constraints. Technical Report 16, University of Strathclyde, 2014.
- [47] P. Rudquist and S.T. Lagerwall. On the flexoelectric effect in nematics. *Liq. Cryst.*, 23:503–510, 1997.
- [48] G.A. Shultz, R.B. Schnabel, and R.H. Byrd. A family of trust-region-based algorithms for unconstrained minimization with strong global convergence properties. *SIAM J. Numer. Anal.*, 22(1):47–67, Feb. 1985.
- [49] D.C. Sorensen. Newton’s method with a model trust-region modification. *SIAM J. Numer. Anal.*, 19:409–426, 1982.
- [50] G. Starke. Gauss-Newton multilevel methods for least-squares finite element computations of variably saturated subsurface flow. *Computing*, 64:323–338, 2000.
- [51] I.W. Stewart. *The Static and Dynamic Continuum Theory of Liquid Crystals: A Mathematical Introduction*. Taylor and Francis, London, 2004.
- [52] K. Su and D. Pu. A nonmonotone filter trust region method for nonlinear constrained optimization. *J. Comput. Appl. Math.*, 223:230–239, 2009.
- [53] D. Thomsen, P. Keller, J. Naciri, R. Pink, H. Jeon, D. Shenoy, and B. Ratna. Liquid crystal elastomers with mechanical properties of a muscle. *Macromolecules*, 34(17):5868–5875, 2001.
- [54] A. Vardi. A trust region algorithm for equality constrained minimization: Convergence properties and implementation. *SIAM J. Numer. Anal.*, 22(3):575–591, 1985.
- [55] E.G. Virga. *Variational Theories for Liquid Crystals*. Chapman and Hall, London, 1994.
- [56] Y. Wan and D. Zhao. On the controllable soft-templating approach to mesoporous silicates. *Chem. Rev.*, 107(7):2821–2860, 2007.
- [57] H. Wu, X. Xu, and C. Liu. On the general Ericksen-Leslie system: Parodi’s relation, well-posedness and stability. *Arch. Rational Mech. Anal.*, pages 59–107, 2013.
- [58] M. Yamada, M. Kondo, J. Mamiya, Y. Yu, M. Kinoshita, C. Barrett, and T. Ikeda. Photomobile polymer materials: Towards light-driven plastic motors. *Angew. Chem. Int.*, 47(27):4986–4988, 2008.